

Copyright © 1997
Hughes Aircraft Company
Unpublished Work

This material may be reproduced by or for the U.S. Government pursuant to the copyright license under the FAR Clause at 52.227-14 (JUN 87)

Document control number: 1721798, Date: 1 July, 1997

Build 4.6

DRAFT

SYSTEM INTERFACE DESIGN DOCUMENT (SIDD)

FOR THE

SURFACE SEARCH RADAR PROGRAM

CONTRACT NO. DTCG23-96-C-ASR009

CDRL SEQUENCE NO. 0017-007

Prepared for:

Commandant (G-ASM/SSR)
US COAST GUARD
2100 Second Street, SW
Washington, DC 20593-0001

Prepared by:

Maritime Systems Program Office
Hughes Naval and Maritime Systems
Building 617, MS B110
P. O. Box 3310
Fullerton, CA 92834-3310

Authenticated by _____
(Contracting agency)

Approved by _____
(Contractor)

Date _____

Date _____

TABLE OF CONTENTS

1. INTRODUCTION	1
2. AN/SPS-73(V) MISSION.....	1
3. SYSTEM OVERVIEW	3
3.1 COMMUNICATORS	3
3.2 SYSTEM CONTROL.....	4
4. SOFTWARE ARCHITECTURE.....	7
4.1 AN/SPS-73(V) SYSTEM INTERFACE	7
4.1.1 <i>AN/SPS-73(V) Inputs.....</i>	7
4.1.2 <i>AN/SPS-73(V) Outputs</i>	7
4.2 AN/SPS-73(V) SYSTEM COMPONENT ARCHITECTURE.....	8
4.3 RADAR GROUP COMPONENTS.....	9
4.3.1 <i>SPDP Unit.....</i>	10
4.3.2 <i>RIPC Unit.....</i>	10
4.3.3 <i>Radar Image Process.....</i>	11
4.4 FUSION PROCESS	11
4.5 NSIU PROCESS	12
4.5.1 <i>Navigational Sensor Interface.....</i>	12
4.5.2 <i>NMEA 0183 Interface</i>	13
4.5.3 <i>Non-NMEA Navigational Interface</i>	13
4.6 OPERATOR POSITION (OP) PROCESS.....	14
5. RDS COMMUNICATIONS METHODOLOGY	15
6. REFERENCE DOCUMENTS	16
6.1 SPECIFICATIONS	16
6.2 STANDARDS	16
6.3 OTHER DOCUMENTS, DRAWINGS, AND PUBLICATIONS	16
7. RDS API MESSAGE CATALOG.....	17
7.1 CATEGORY: ANY OWNSHIP REPORT.....	17
7.2 CATEGORY: ANY STATUS	17
7.3 CATEGORY: ANY ALERT	18
7.4 CATEGORY: ANY TRACK DATA	18
7.5 CATEGORY: BIT TEST REQUEST.....	19
7.6 CATEGORY: BIT TEST STATUS	19
7.7 CATEGORY: FUSION TO RP REQUEST.....	20
7.8 CATEGORY: FUSION TO OP ANY STATUS	20
7.9 CATEGORY: ICC ASSOCIATION	20
7.10 CATEGORY: INTERCONNECTION CONTROL	21
7.11 CATEGORY: NETWORK CONTROL.....	22
7.12 CATEGORY: NSIU ANY STATUS	22
7.13 CATEGORY: OP TO FUSION OPERATOR ENTRY	22
7.14 CATEGORY: OP GRAPHIC DEFINITION.....	23
7.15 CATEGORY: OP TO NSIU OPERATOR ENTRY	23
7.16 CATEGORY: OP TO RCC OPERATOR ENTRY	24
7.17 CATEGORY: OP TO RDE OPERATOR ENTRY	24

7.18	CATEGORY: OP TO RP OPERATOR ENTRY	25
7.19	CATEGORY: OP TO RSC OPERATOR ENTRY.....	26
7.20	CATEGORY: RADAR IMAGE DATA	26
8.	API MESSAGE FORMATS	27
8.1	RADAR IMAGE MESSAGE (PRIMARY & SECONDARY FORMAT).....	28
8.2	API MESSAGE LISTINGS BY AMP TYPE	29
9.	ATTACHMENT: RDS API HEADER FILES	69

ILLUSTRATIONS

FIGURE 1.	USCGC HAMILTON CONFIGURATION.....	1
FIGURE 2.	WAGB-400 CONFIGURATION EXAMPLE	3
FIGURE 3.	WPB-110 SYSTEM CONFIGURATION.....	4
FIGURE 4.	INTERCONNECTION CONTROL CONFIGURATION	5
FIGURE 5.	AN/SPS-73(V) CONTEXT DIAGRAM.....	7
FIGURE 6.	AN/SPS-73(V) SYSTEM COMMUNICATIONS MODEL.....	8
FIGURE 7.	RADAR GROUP COMPONENTS	9
FIGURE 8.	RADAR PROCESSOR FUNCTIONAL MODEL.....	10
FIGURE 9.	FUSION PROCESS.....	11
FIGURE 10.	NSIU FUNCTIONAL MODEL.....	12
FIGURE 11.	OPERATOR POSITION PROCESS.....	14
FIGURE 12.	RDS COMMUNICATION MODEL	15

TABLES

TABLE 1.	CAT_ANY_OWNERSHIP_REPORT	17
TABLE 2.	CAT_ANY_STATUS	17
TABLE 3.	CAT_ANY_ALERT	18
TABLE 4.	CAT_ANY_TRACK_DATA	18
TABLE 5.	CAT_BIT_TEST_REQUEST.....	19
TABLE 6.	CAT_BIT_TEST_STATUS	19
TABLE 7.	CAT_FUSION_OP_ANY_STATUS	20
TABLE 8.	CAT_ICC_ASSOCIATION	20
TABLE 9.	CAT_INTERCONNECTION_CONTROL	21
TABLE 10.	CAT_NSIU_ANY_STATUS	22
TABLE 11.	CAT_OP_FUSION_OPERATOR_ENTRY	22
TABLE 12.	CAT_OP_GRAPHIC_DEFINITION	23
TABLE 13.	CAT_OP_NSIU_OPERATOR_ENTRY	23
TABLE 14.	CAT_OP_RCC_OPERATOR_ENTRY	24
TABLE 15	CAT_OP_RDE_OPERATOR_ENTRY	24
TABLE 16.	CAT_OP_RP_OPERATOR_ENTRY	25
TABLE 17.	CAT_OP_RSC_OPERATOR_ENTRY.....	26
TABLE 18 :	CAT_RADAR_x_ANY_IMAGE (x = 1 TO 4).....	26
TABLE 19.	API MESSAGE HEADER STRUCTURE.....	27
TABLE 20.	AMP_ANY_IMAGE (WHERE ANY CAN BE {PRIMARY SECONDARY})	28
TABLE 21.	AMP_ACKNOWLEDGE (ICC_ACKNOWLEDGE_MSG)	29

TABLE 22. AMP_ADF_MESSAGE (ADF_RECORD)	29
TABLE 23. AMP_ASSIGN_LEVEL_ONE_CONTROL (ICC_ASSIGN_LEVEL_ONE_CONTROL_MSG)	30
TABLE 24. AMP_ASSIGN_LEVEL_TWO_CONTROL (ICC_ASSIGN_LEVEL_TWO_CONTROL_MSG)	31
TABLE 25. AMP_ASSIGN_NSIU_CONTROL (ICC_ASSIGN_NSIU_CONTROL_MSG)	32
TABLE 26. AMP_BP_HEARTBEAT (BP_HEARTBEAT_MSG)	32
TABLE 27. AMP_DISSOLVE (ICC_DISSOLVE_MSG)	32
TABLE 28. AMP_EXECUTE_DEFUSE_SYSTEM_TRACK (SYSTEM_TRACK_DEFUSE_REQUEST)	32
TABLE 29. AMP_EXECUTE_DROP_SYSTEM_TRACK (SYSTEM_TRACK_DROP_RECORD)	33
TABLE 30. AMP_EXECUTE_FUSE_SYSTEM_TRACK (SYSTEM_TRACK_FUSE_REQUEST)	33
TABLE 31. AMP_EXECUTE_MANUAL_SENSOR_TRACK_ACQUIRE (TRACK_ACQUIRE_REQUEST)	33
TABLE 32. AMP_EXECUTE_RESET (RP_RESET_EXECUTE_MSG)	34
TABLE 33. AMP_FUSION_DISSOLVE (ICC_FUSER_DISSOLVE_MSG)	34
TABLE 34. AMP_FUSION_IDENTIFICATION (ICC_FUSER_ASSOCIATION_MSG)	34
TABLE 35. AMP_GENERIC_RADAR_ORIENTATION (RP_GENERIC_RADAR_ORIENTATION_MSG)	34
TABLE 36. AMP_GRAPHIC_CIRCLE (GRAPHIC_MESSAGE)	35
TABLE 37. AMP_GRAPHIC_COMMAND (GRAPHIC_COMMAND_RECORD)	35
TABLE 38. AMP_GRAPHIC_POLYLINE (GRAPHIC_MESSAGE)	36
TABLE 39. AMP_ICC_HEARTBEATS (ICC_HEARTBEAT_MSG)	37
TABLE 40. AMP_IMAGE_AVAILABILITY (ICC_IMAGE_AVAILABILITY_MSG)	37
TABLE 41. AMP_LAND_MASK_MAP_DOWNLOAD (LAND_MASK_DOWNLOAD_RECORD)	38
TABLE 42. AMP_LAND_MASK_SELECT (LAND_MASK_SELECT_RECORD)	38
TABLE 43. AMP_LEVEL_ONE_PARAMETERS (ICC_LEVEL_ONE_PARAMETERS_MSG)	38
TABLE 44. AMP_LEVEL_TWO_PARAMETERS (ICC_LEVEL_TWO_PARAMETERS_MSG)	39
TABLE 45. AMP_MANUAL_OWNSHIP_REPORT (MANUAL_OWNSHIP_REPORT_MSG)	39
TABLE 46. AMP_NAV_SENSOR_STATUS_REQUEST (NAV_SENSOR_STATUS_MSG)	40
TABLE 47. AMP_NAV_SENSOR_STATUS_RESPONSE (NAV_SENSOR_STATUS_MSG)	40
TABLE 48. AMP_NSIU_DISSOLVE (ICC_NSIU_DISSOLVE_MSG)	40
TABLE 49. AMP_NSIU_IDENTIFICATION (ICC_NSIU_ASSOCIATION_MSG)	40
TABLE 50. AMP_PAUSE_RADAR_CONTROL (ICC_RADAR_CONTROL_MSG)	41
TABLE 51. AMP_RADAR_IMAGE_PRESENTATION (RP_IMAGE_PRESENTATION_SELECT_MSG)	41
TABLE 52. AMP_RADAR_SHUTDOWN (ICC_RADAR_SHUTDOWN_MSG)	41
TABLE 53. AMP_RADAR_TIMER_COUNTDOWN (RP_TIMER_COUNTDOWN_MSG)	41
TABLE 54. AMP_RCC_IDENTIFICATION (ICC_RCC_ASSOCIATION_MSG)	41
TABLE 55. AMP_RCC_STATUS (RADAR_STATUS_MSG)	42
TABLE 56. AMP_RDP_RIPC_SPO (RP_RIPC_SPO_MSG)	43
TABLE 57. AMP_RDP_RSC_SPO (RP_RSC_SPO_MSG)	43
TABLE 58. AMP_RDP_SCAN_STATUS (RP_SCAN_REPORT_MSG)	43
TABLE 59. AMP_RDP_SPDP_SPO (RP_SPDP_SPO_MSG)	44
TABLE 60. AMP_RDP_STATUS_OUTPUT (RP_STATUS_MSG)	44
TABLE 61. AMP_ROLE_ASSIGNMENT_TABLE (ICC_ROLE_ASSIGNMENT_TABLE_MSG)	44
TABLE 62. AMP_RP_IDENTIFICATION (ICC_RP_ASSOCIATION_MSG)	45
TABLE 63. AMP_RSC_IDENTIFICATION (ICC_RSC_ASSOCIATION_MSG)	45
TABLE 64. AMP_SELECT_AGC_TOGGLE (RP_AGC_SELECT_MSG)	45
TABLE 65. AMP_SELECT_AUTO_MODE (RP_AUTO_MODE_SELECT_MSG)	45
TABLE 66. AMP_SELECT_BLANKING (BLANKING_SELECT_MSG)	46
TABLE 67. AMP_SELECT_ENUM_RANGE_SCALE (ENUM_RANGE_SCALE_SELECT_MSG)	46
TABLE 68. AMP_SELECT_FILTERING (FILTER_SELECT_MSG)	46
TABLE 69. AMP_SELECT_IR_FALL (IR_FALL_SELECT_MSG)	47
TABLE 70. AMP_SELECT_IR_RISE (IR_RISE_SELECT_MSG)	47
TABLE 71. AMP_SELECT_LANDMASK_VIEW (RP_LAND_MASK_SELECT_MSG)	47
TABLE 72. AMP_SELECT_MANUAL_MODE (RP_MANUAL_MODE_SELECT_MSG)	47
TABLE 73. AMP_SELECT_NEAR_LAND (NEAR_LAND_MODE_SELECT_MSG)	48

TABLE 74. AMP_SELECT_ORIENTATION (RP_SELECT_ORIENTATION_MSG).....	48
TABLE 75. AMP_SELECT_PM (PERFORMANCE_MONITOR_SELECT_MSG)	48
TABLE 76. AMP_SELECT_PULSE_IR (RP_PULSE_IR_SELECT_MSG).....	48
TABLE 77. AMP_SELECT_RADAR_STATE (RP_STATE_SELECT_MSG).....	49
TABLE 78. AMP_SELECT_RANGE_SCALE (RP_RANGE_SCALE_SELECT_MSG)	49
TABLE 79. AMP_SELECT_STC_TOGGLE (RP_STC_SELECT_MSG).....	49
TABLE 80. AMP_SELECT_SYSTEM_TRACK_AFFIL (SYSTEM_TRACK_AFFIL_SELECT_MSG)	49
TABLE 81. AMP_SELECT_SYSTEM_TRACK_CLASS (SYSTEM_TRACK_CLASS_SELECT_MSG).....	50
TABLE 82. AMP_SELECT_TRACK_CLASS (SYSTEM_TRACK_CLASS_SELECT_MSG).....	50
TABLE 83. AMP_SELECT_VIDEO_SOURCE (RCC_VIDEO_SELECTION_MSG).....	50
TABLE 84. AMP_SENSOR_TRACK_DROP (TRACK_DROP_RECORD).....	51
TABLE 85. AMP_SENSOR_TRACK_LOST (TRACK_LOST_RECORD)	51
TABLE 86. AMP_SENSOR_TRACK_REPORT (SENSOR_TRACK_TYPE).....	52
TABLE 87. AMP_SET_ACP_THRESHOLD (RP_ACP_THRESHOLD_SET_MSG).....	53
TABLE 88. AMP_SET_ARP_THRESHOLD (RP_ARP_THRESHOLD_SET_MSG)	53
TABLE 89. AMP_SET_AUTO_RF_TUNE (RF_TUNING_SELECT_MSG).....	54
TABLE 90. AMP_SET_BIAS_CORRECTION (RP_AZIMUTH_BIAS_SET_MSG).....	54
TABLE 91. AMP_SET_DECAY_RATE (RP_DECAY_RATE_SET_MSG).....	54
TABLE 92. AMP_SET_FTC (RP_FTC_SET_MSG)	54
TABLE 93. AMP_SET_GUARD_CELLS (RP_GUARD_CELLS_SET_MSG).....	55
TABLE 94. AMP_SET_MANUAL_RF_TUNE (RF_TUNING_CONTROL_MSG).....	55
TABLE 95. AMP_SET_MISSION (RP_MISSION_MODE_SELECT_MSG).....	55
TABLE 96. AMP_SET_ORIGIN_OFFSET (RP_ORIGIN_OFFSET_SET_MSG).....	56
TABLE 97. AMP_SET_RADAR_IMAGE (RADAR_IMAGE_SELECT_MSG).....	56
TABLE 98. AMP_SET_RADAR_MAIN_BANG (MAIN_BANG_CONTROL_MSG)	56
TABLE 99. AMP_SET_RADAR_PARAMETERS (RP_PARAMETERS_MSG).....	57
TABLE 100. AMP_SET_RANGE_BIAS (RP_RANGE_BIAS_SET_MSG)	57
TABLE 101. AMP_SET_RCC_PARAMETERS (RCC_PARAMETERS_MSG).....	57
TABLE 102. AMP_SET_RP_PARAMETERS (RP_PARAMETERS_MSG).....	59
TABLE 103. AMP_SET_RSC_PARAMETERS (RSC_PARAMETERS_MSG).....	59
TABLE 104. AMP_SET_SCALER_VALUE (RP_RVF_SCALER_SET_MSG).....	60
TABLE 105. AMP_SET_SENSOR_TRACK_SPEEDS (TRACK_SPEED_FILTER_SET_MSG).....	61
TABLE 106. AMP_SET_STC_CURVE (RP_STC_SET_CURVE_MSG).....	61
TABLE 107. AMP_SET_SYSTEM_TIME (SYSTEM_TIME_SET_RECORD)	61
TABLE 108. AMP_SET_SYSTEM_TRACK_LABEL (SYSTEM_TRACK_LABEL_SET_MSG)	61
TABLE 109. AMP_SET_TBM_BIASES (RP_TBM_BIASES_SET_MSG).....	62
TABLE 110. AMP_SET_THRESHOLD_FORMAT (RP_THRESHOLD_FORMAT_SET_MSG)	62
TABLE 111. AMP_SET_TRACK_SCAN_COUNT (TRACK_SCAN_COUNT_SET_MSG)	62
TABLE 112. AMP_SET_TRIG_THRESHOLD (RP_TRIG_THRESHOLD_SET_MSG)	62
TABLE 113. AMP_SET_VIDEO (RP_VIDEO_SET_MSG).....	63
TABLE 114. AMP_SETUP_DATA_1 (ICC_SETUP_DATA_MSG_1)	63
TABLE 115. AMP_SETUP_DATA_2 (ICC_SETUP_DATA_MSG_2)	63
TABLE 116. AMP_SIU_STATUS_CONTROL_REQUEST (SIU_STATUS_CONTROL_MSG)	63
TABLE 117. AMP_SIU_STATUS_CONTROL_RESPONSE (SIU_STATUS_CONTROL_MSG)	64
TABLE 118. AMP_SPDP_IDENTIFICATION (ICC_SPDP_ASSOCIATION_MSG)	64
TABLE 119. AMP_SYSTEM_ALERT (SYSTEM_ALERT_MSG).....	64
TABLE 120. AMP_SYSTEM_OWNERSHIP_REPORT (SYSTEM_OWNERSHIP_REPORT_MSG)	65
TABLE 121. AMP_SYSTEM_TRACK_DROP (SYSTEM_TRACK_DROP_RECORD)	65
TABLE 122. AMP_SYSTEM_TRACK_REPORT (SYSTEM_TRACK_RECORD)	65
TABLE 123. AMP_SYSTEM_TRACK_REPORT CONT... (SYSTEM_TRACK_TYPE).....	66
TABLE 124. AMP_TARGET_DETECTION_REPORT (TARGET_DETECTION_RECORD)	67
TABLE 125. AMP_TARGET_REPORT (TRACK_TARGET_REPORT).....	68

ACRONYM LIST

<u>Acronym</u>	<u>Expansion</u>
ANSI	American National Standards Institute
API	Application Program Interface
BIT	Built-In-Test
CO	Control Manager
ETS	External Tactical System
FDDI	Fiber Distributed Data Interface
FTC	Fast Time Constant
ICC	InterConnection Control
IEEE	Institute of Electrical and Electronic Engineers
IP	Internet Protocol
LAN	Local Area Network
MMI	Man-Machine Interface
NMEA	National Marine Electronics Association
NSIU	Navigation Sensor Interface Unit
OP	Operator Position
RCC	Radar Control and Converter
RDE	Radar Data Extractor
RDS	Radar Data Services
RI	Radar Input (process)
RPC	Radar data processor Interface and Peripheral Controls
RP	Radar Processor
RSC	Radar Scan Converter
SAOP	Stand Alone Operator Position
SD	Situation Display (process)
SPDP	Signal Processor control and Detection Processing
SPS	Shipboard Pulsed System
SSR	Surface Search Radar
STC	Sensitivity Time Control
TBOP	Table-top / Bulkhead mounted Operator Position
TCP	Transmission Control Protocol
TM	Track Manager
UDP	User Datagram Protocol
VME	VERSA-Module Europe

1.

1. Introduction

The purpose of this document is to describe the external and internal interfaces for the Hughes Surface Search Radar System, AN/SPS-73(V). The document also describes the content of the Radar Data Service (RDS) Applications Programming Interface (API) messages. The RDS API message set provides the interoperability between the AN/SPS-73 software processes and will also be used by External Tactical Systems, herein referred to as an ETS.

2. AN/SPS-73(V) Mission

The AN/SPS-73(V) will provide radar surface search and tracking services for the US Coast Guard, US Navy, and Vessel Traffic Systems. The AN/SPS-73(V) system will consist of a minimum of 16 different configurations, each containing varying quantities of Stand Alone Operator Positions (SAOP console), Table-Top/Bulkhead mounted Operator Positions (TBOP console), and Furuno X and S band radars. The AN/SPS-73(V) system will provide indications of surface craft, obstructions, shorelines, buoys, and navigation marks in a manner that will assist in navigation and vessel tracking. The AN/SPS-73(V) system will be capable of tracking and continuously updating up to 200 tracks per radar and to display up to 300 tracks per operator position.

In addition, multiple external platform interfaces are supported such as, MK-92, AN/SPS-67, and AN/SPS-55 radars, and ship navigational equipment, such as Gyrocompass, Speed Log, Loran, and GPS/dGPS. The system supports up to 16 NMEA 0183 ports.

For instance, the US Coast Guard's Hamilton (WHEC-715) configuration¹ as depicted in Figure 1, a 378-Foot High Endurance Cutter commissioned in 1967, is characterized by two Stand Alone Operator Positions, and one Table-Top/Bulkhead mounted Operator Position, separate Furuno 25kW X-band and 30kW S-band antennas, as well as an *external tactical system* (ETS console) all connected via a dual FDDI ring network.

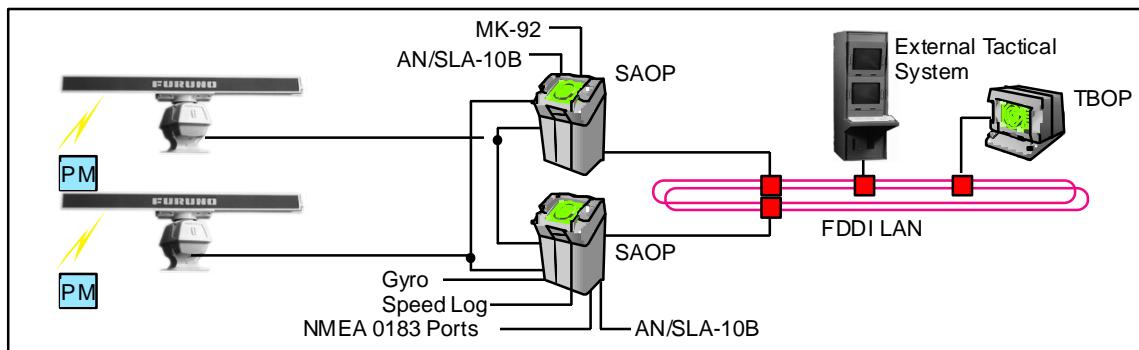


Figure 1. USCGC Hamilton configuration.

When only two systems, such as a SAOP and ETS, are interconnected, an Ethernet LAN is used (as depicted in Figure 3). The USCGC Patrol Boat WPB-110 is an example of such a configuration, which is composed of one SAOP with *radar processor* (RP) connected to a Furuno 25kW X-band antenna, and an External Tactical System.

¹ Note: Figure 1, depicts the 378 class ship configurations. The initial Hamilton configuration will not include an External Tactical System.

3.

System Overview

Within any AN/SPS-73(V) system configuration, one or two SAOPs will contain a Radar Processor (RP) unit. Each RP can only select one radar video source at any given time. Figure 2 depicts such a configuration for a WAGB-400 class ship. The radar video is used for both the synthesis of track reports and the display of digitized radar images.

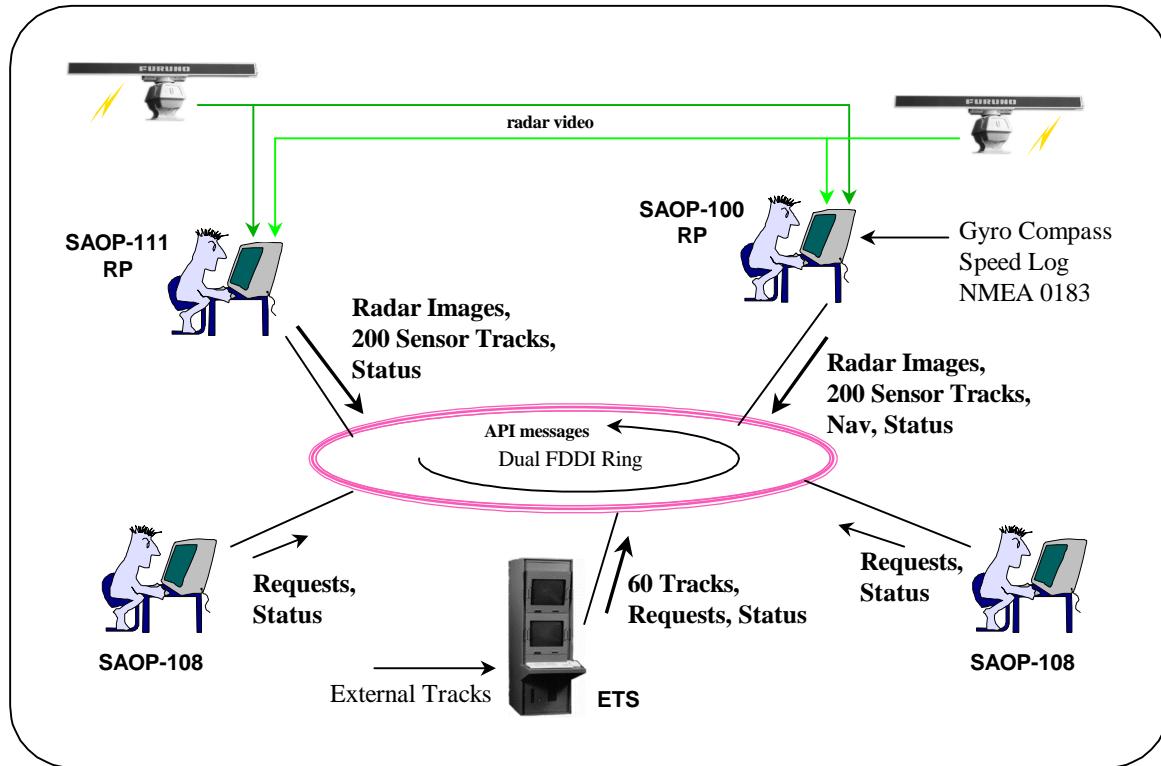


Figure 2. WAGB-400 Configuration Example.

3.1 Communicators

Each RP will provide sensor track report messages (each RP can track up to 200 tracks), in addition to primary/secondary radar images from the selected radar. Note all [API] messages that appear on the network are distributed via the RDS (Radar Data Services) Communications software. Also, all consoles transmit periodic messages describing their status via heartbeat messages. The remaining consoles (SAOPs, TBOPs, or ETS²) will request to receive these track report³ messages, status indications, and any one of the radar image messages. Navigational data messages (derived from gyrocompass, speed log, or NMEA ports) will be provided by only one SAOP per system configuration. Most configurations use the FDDI dual ring topology.

² Within the context of SSR, ETS consoles are functionally equivalent to TBOPs however – architecturally – very different.

³ Track report messages, in this context, refer to System Track Reports, which are derived from sensor tracks through a system level correlation process called *fusion*.

As a product feature, for systems that only contain two consoles, an Ethernet topology is used to reduce overall system cost; this configuration is depicted in Figure 3, for a WPB-110 class system.

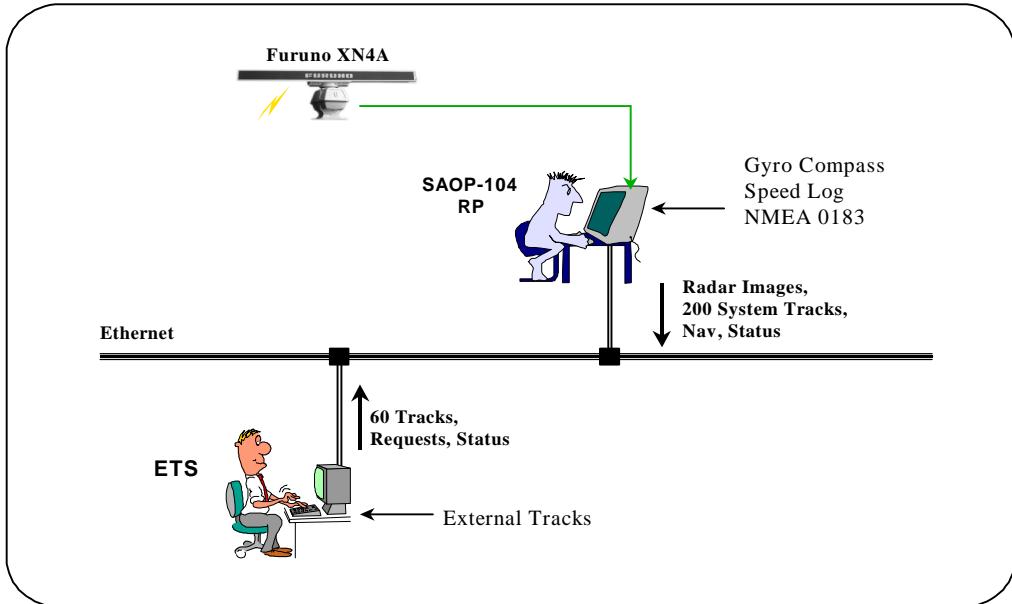


Figure 3. WPB-110 system configuration

The AN/SPS-73(V) Application Program Interface is message based and supports the registration of message requests, receiving these messages, and the transmission of requested messages. The message types are defined in this document. Another way of thinking is that the messaging system could be thought of as an *API Highway*. The API Highway provides a generic communications metaphor, in which messages don't care how they are transported, as long as they get there; the underlying code provides a vehicle to support the delivery of these messages, whether locally (same CPU), over the VME backplane, or over an attached network.



3.2 System Control

System control (*InterConnection Control*) is accomplished by assigning to an operator position various levels of control of the radar information processing and navigational sensor overrides. When we say that a particular operator position (OP), including the ETS, has control over a radar or the navigation process, we are referring to a pairing between OPs and a service provider⁴ (as depicted in Figure 4) which could exist within a different enclosure. The following control levels are defined:

1. Level 1 controllers have physical control, via a Radar Control and Converter (RCC) card, of one Furuno radar⁵, the associated Radar Data Extractor (RDE) card settings, and the primary radar image scan converter

⁴ The term *service provider* refers collectively to the RCC, RP, RDE, RSC tasks within each RP system.

⁵ Additionally the RP SAOPs are designed to observe but not control, three generic radar types: the MK-92, AN/SPS-55, and AN/SPS-67 radars.

(RSC) card. Thus, the operator can control the range scale of the primary radar image and the output format, either Cartesian or Rho/Theta. Furthermore, radar control includes setting the pulse width, pulse repetition frequency, STC (sensitivity time control), FTC (fast time constant), gain, tuning, non-transmit zones, and standby control.

2. Level 2 controllers can only control the offset and range scale of the secondary radar image, the output format of Cartesian or Rho/Theta, and local STC and FTC functions.
3. Level 3 (user), is a non-controlling destination for the radar image, and therefore has no control over the radar image produced by the Radar Scan Converters in the Radar Processor. Level 3 viewers can do local zooming of the selected radar image.
4. To control the navigation process (NSIU), one designated OP will be allowed to specify which manual navigational inputs will be used for ownship position parameters. Also, COMDAC ownship reports will be accepted from the ETS.
5. One SAOP will be designated to become the source of System Track Reports (Fusion process).

Figure 4 provides an example of InterConnection Control (after role assignments have been distributed), wherein a TBOP is controlling one radar video source and an ETS is controlling the other radar video source. The operator at the TBOP console can select between either the Furuno radar video source or the MK-92 radar video source. The SAOP operator can select either of RP1 or RP2 secondary radar images.

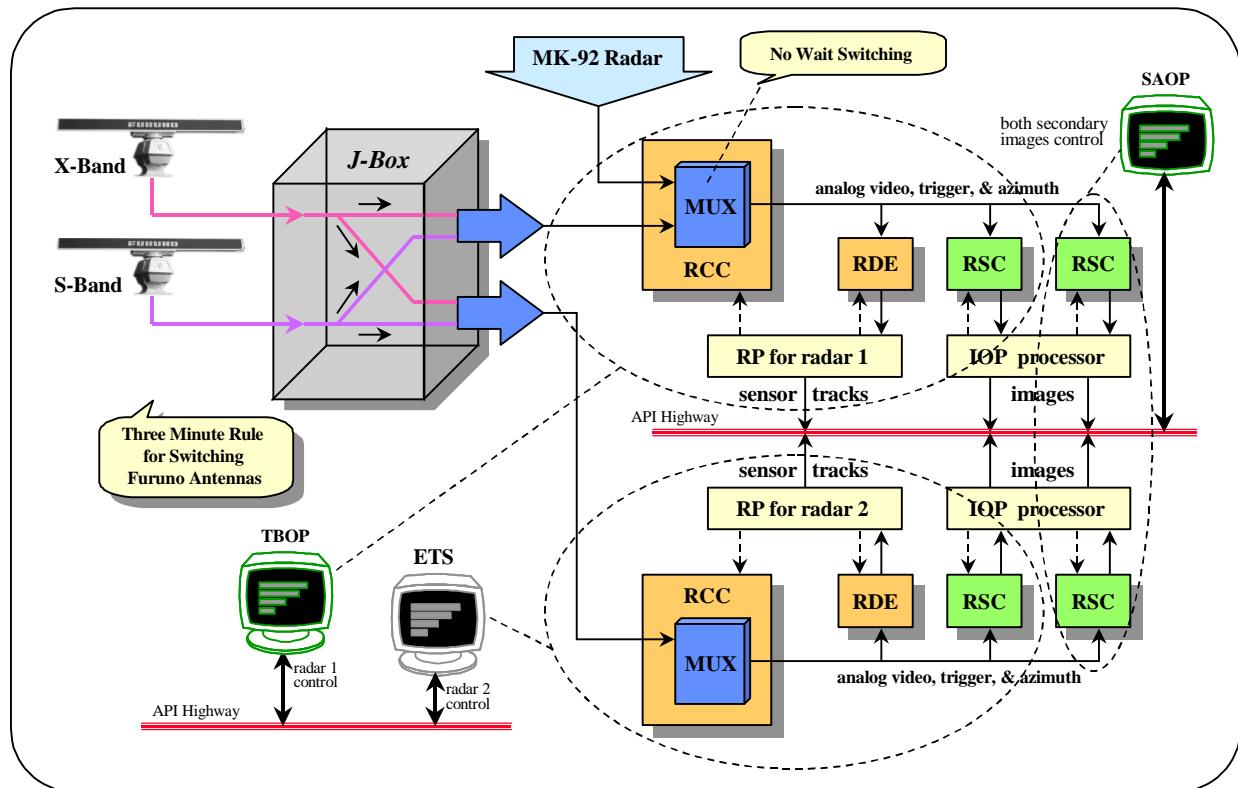


Figure 4. InterConnection Control Configuration.

4.

Software Architecture

For purposes of clarity, when describing the software architecture, the AN/SPS-73(V) system should be viewed as a single entity. The following context diagram, Figure 5, presents the expected inputs and outputs of the AN/SPS-73(V) system software.

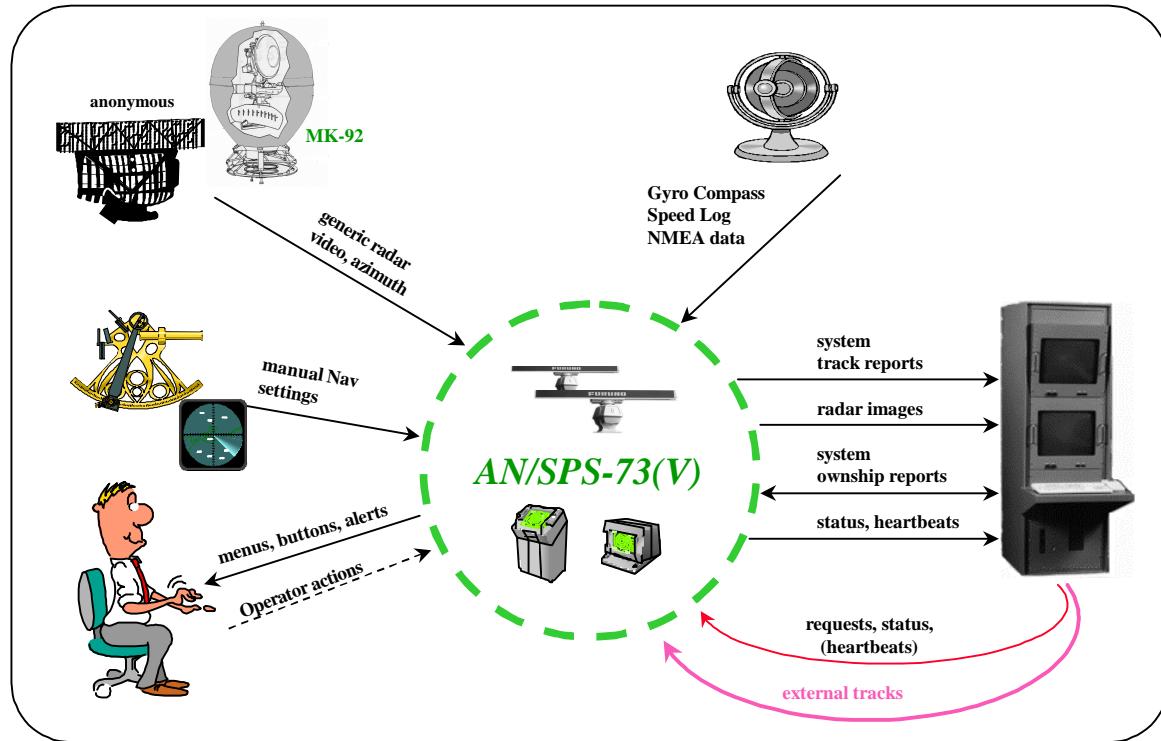


Figure 5. AN/SPS-73(V) Context Diagram.

4.1 AN/SPS-73(V) System Interface

4.1.1 AN/SPS-73(V) Inputs

AN/SPS-73(V) inputs are comprised of: operator actions, manual navigational settings, *Selected Radar* video & azimuth, gyro compass, speed logs, and sensor inputs via NMEA 0183 strings, requests for image/track reports, and heartbeats (health/status). Also additional inputs can include External Track Reports and SCCS Navigational (COMDAC Ownership) Reports.

4.1.2 AN/SPS-73(V) Outputs

AN/SPS-73(V) outputs include: Navigational (Ownership) Reports, System (*fused*) Track Reports, Radar Images (primary and secondary image for each radar), status/heartbeats (periodic health messages).

4.2

AN/SPS-73(V) System Component Architecture

The major components within the AN/SPS-73(V) system architecture, as depicted in Figure 6, are: one or two *Radar Groups* with their associated *J-Box*, a *Fusion process*, and an *NSIU service* process. Each of these components provide outputs which are consumed by local and remote operator positions (OP and ETS). All messaging inputs and outputs for the system, over the API Highway, are handled through the *RDS API*.

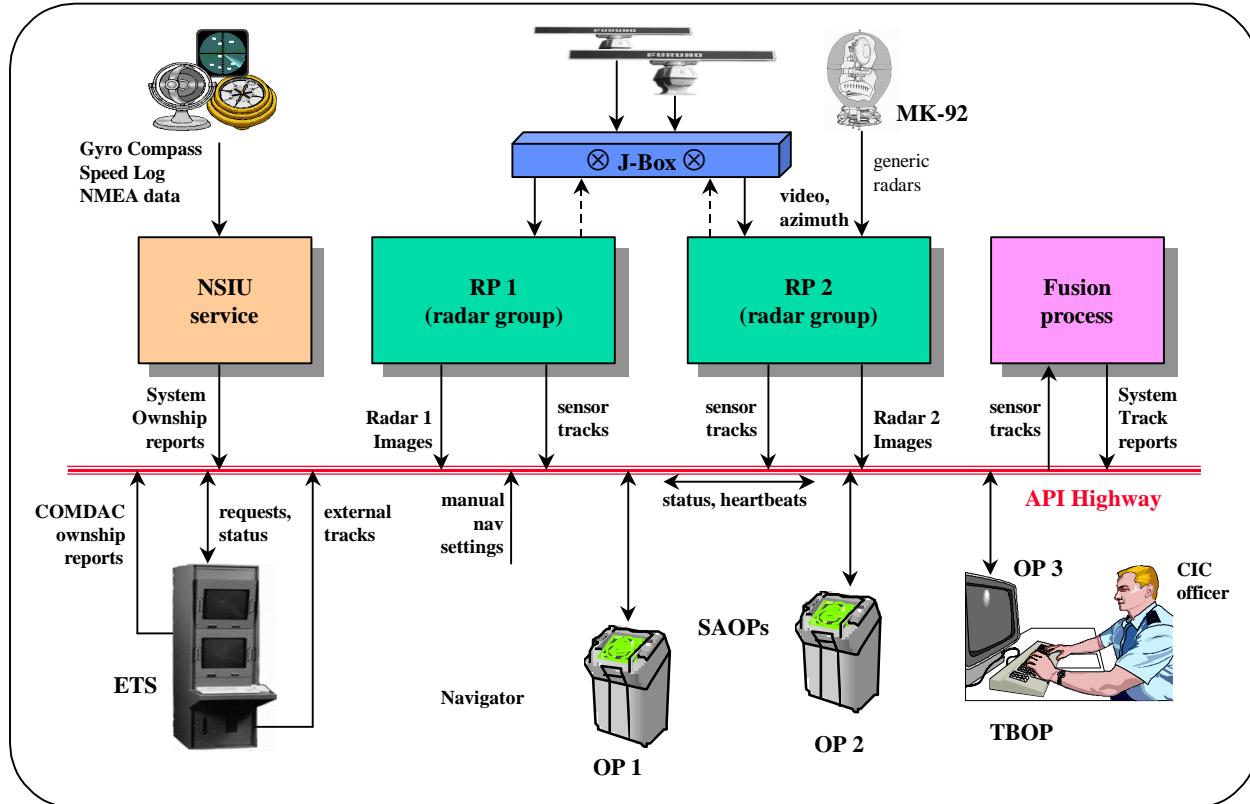


Figure 6. AN/SPS-73(V) System Communications Model

The *NSIU service process* converts external gyrocompass, speed logs, and NMEA data into navigational reports, called “System Ownship Reports”. The *Radar Groups* convert radar video and azimuth values into *sensor track* reports. The *Fusion Process* combines sensor track reports into (*fused*) *System Track Reports*. In addition, the *Radar Groups* transform analog radar video images (primary and secondary) into viewable pixmap formats. The *Operator Position process* (OP) accepts external status requests, system track reports, radar image, heartbeats, status, navigational reports, and operator actions, and outputs status requests, heartbeats, status, control of the system components, and renders graphics, text, and radar images onto the display.

4.3

Radar Group Components

The *Radar Group*, depicted in Figure 7, is composed of two major components: the Radar Processor and the IOP (input/output processing) processor. Several addition hardware components provide external interfaces to the system. The hardware components consist of a Radar Control and Converter card (RCC), a Radar Data Extractor card (RDE), and two Radar Scan Converter cards (RSC).

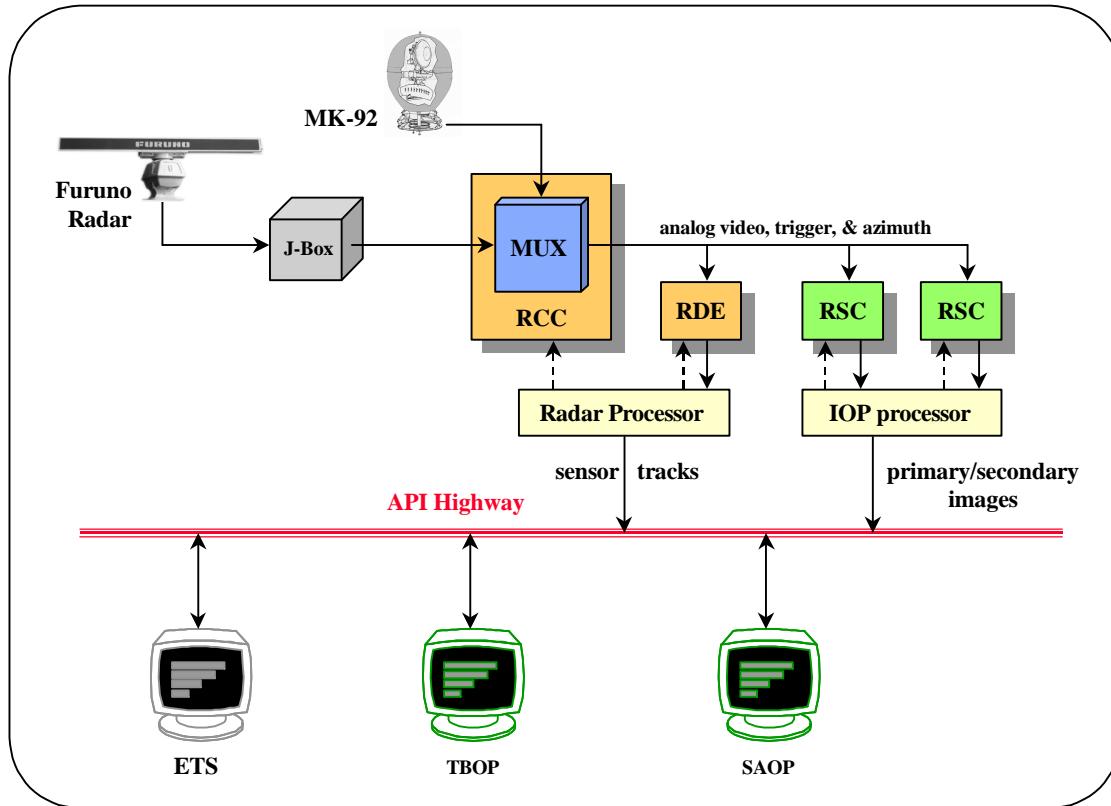


Figure 7. Radar Group Components

Within each *Radar Processor* are two major processes, the SPD⁶ (Signal Processor Control and Detection Processing) Unit and the RIPC (Radar Data Processor Interface and Peripheral Control) Unit. Additionally, an RCC (Radar Control and Converter) task manages the interface to the Furuno radars and generic radar inputs. The Radar Image Process executes within the *IOP processor*.

4.3.1

⁶ The RCC card provides a switching function that connects one of three radar input ports to one RCC output port. One output port provides video, trigger, and azimuth data to the RSC and RDE cards which independently process radar data in parallel. Each RSC and RDE card provides independent (GAIN RSC only), STC, and FTC processing of non-Furuno and Furuno radar video.

SPDP Unit

As shown in Figure 8, the processing of radar information is controlled by the operator via *RDS API* messages. The messages are generated by the OP process in response to operator actions, and are received by *RdsMsg*. Radar detections are converted into sweep clusters by the *RdeData* process. The sweep clusters are combined with azimuth values to produce sector crossings and centroids. Ownship messages from the NSIU Unit are used to enhance the output.

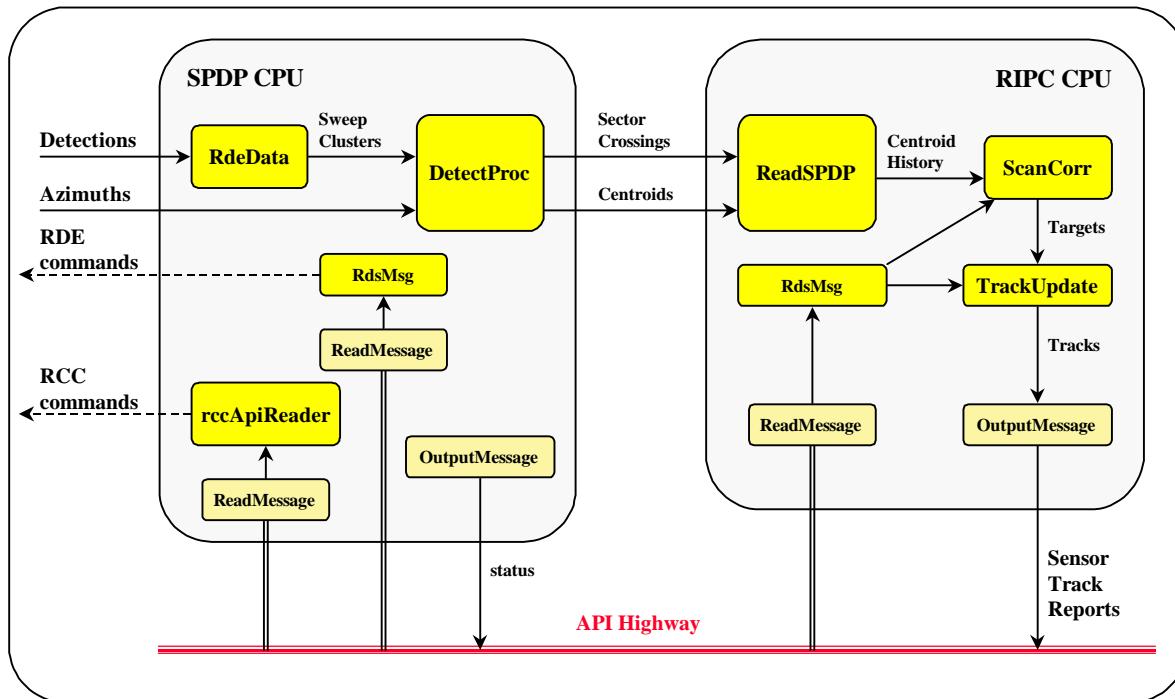


Figure 8. Radar Processor Functional Model

Mini Glossary:

- Detections – Radar sensitivity and gain controls are used to set detection threshold values. *A detection* refers to a radar echo that exceeds the detection threshold. Each detection is assigned an azimuth and a range.
- Centroids – *Centroids* are clusters, or groupings, of detections that are believed to be from a single target as a result of cluster processing.
- Targets – *Targets* are centroids that have been subjected to clutter rejection processing. Clutter rejection is used to distinguish vessel radar echoes from echoes caused by sea surface waves and spray. This is accomplished by comparing centroids from the current radar scan with centroids from the previous radar scan.
- Tracks – *Tracks* are kinematic models derived from targets. The AN/SPS-73(V) compares target positions from the current radar scan with target positions of the previous radar scans to estimate current vessel course and speed.

4.3.2 RIPC Unit

The *ReadSPDP* process maintains a centroid history catalog from sweep clusters and centroids received during each antenna rotation (scan). The *ScanCorr* process uses the centroid history to produce target reports. *TrackUpdate* consumes target reports to generate sensor track reports.

4.3.3 Radar Image Process

The *Radar Image Process*, on the IOP processor, handles the acquisition and distribution of the radar image from the *Radar Scan Converter* cards. Initially, the digitized radar image from the RSC is transferred over the VME bus into a local buffer. If necessary the image is forwarded to other consoles via the RDS communication services.

4.4 Fusion Process

The *Fusion Process*, depicted in Figure 9, is a blending of local and external sensor track reports. The resultant outputs are System Track Reports. System tracks are those output by the fusion process, and include single (unfused) sensor tracks, fused tracks from one sensor or fused tracks from more than one sensor. The technical proposal states: “The proposed HSSR-2000 [sic] system provides automatic multi-source correlation to reduce the operator workload and provide a single consistent situation display”.

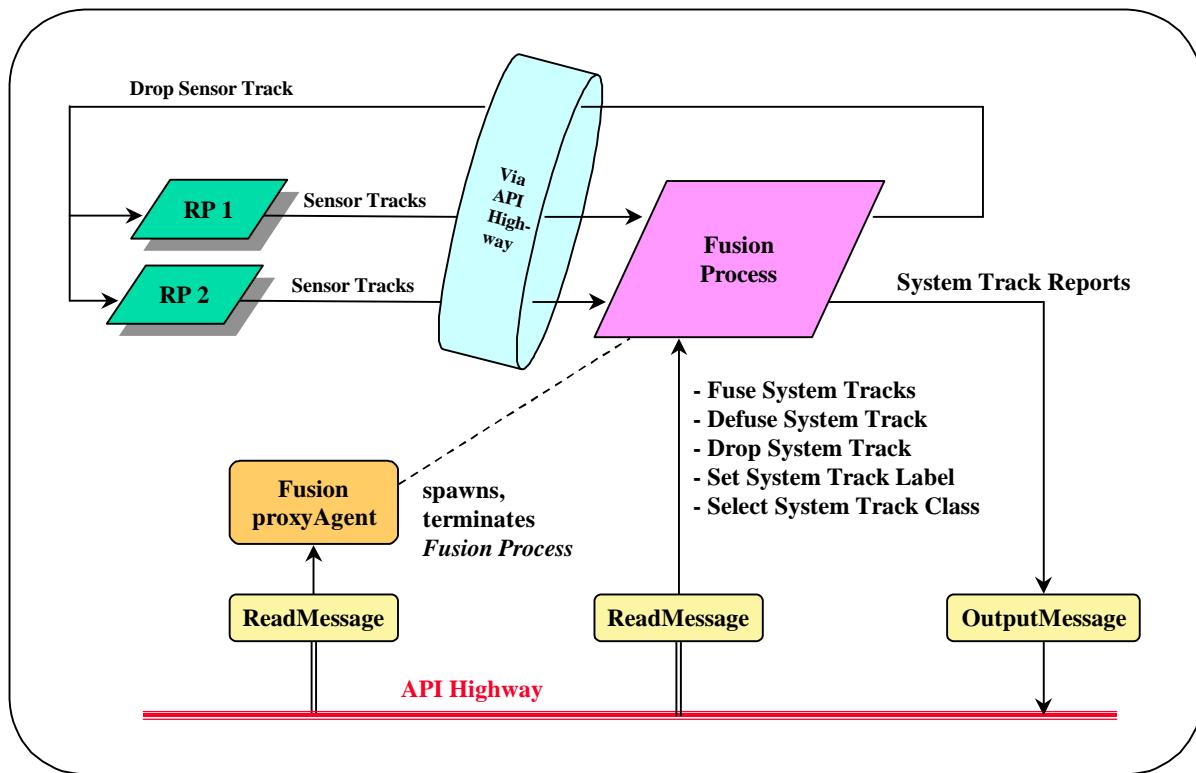


Figure 9. Fusion Process

Only one Fusion process is active at any time. The operator, through InterConnection Control, designates which ‘physical’ console performs this process. For further detail, see the *InterConnection Control Concept Paper*.

4.5

NSIU Process

The NSIU Process, depicted in Figure 10, accepts external navigation sensor inputs from both a Sensor Interface Unit (SIU) VME card and an attached Synergy EU20 Module daughter card. The data received from navigational sensors (e.g., speed log, gyrocompass, GPS, Loran-C, etc.) by the NSIU Process is converted into System Ownship Report messages and sent via the RDS API. This report consists of position data, course data, heading data, ground speed data, and water speed data.

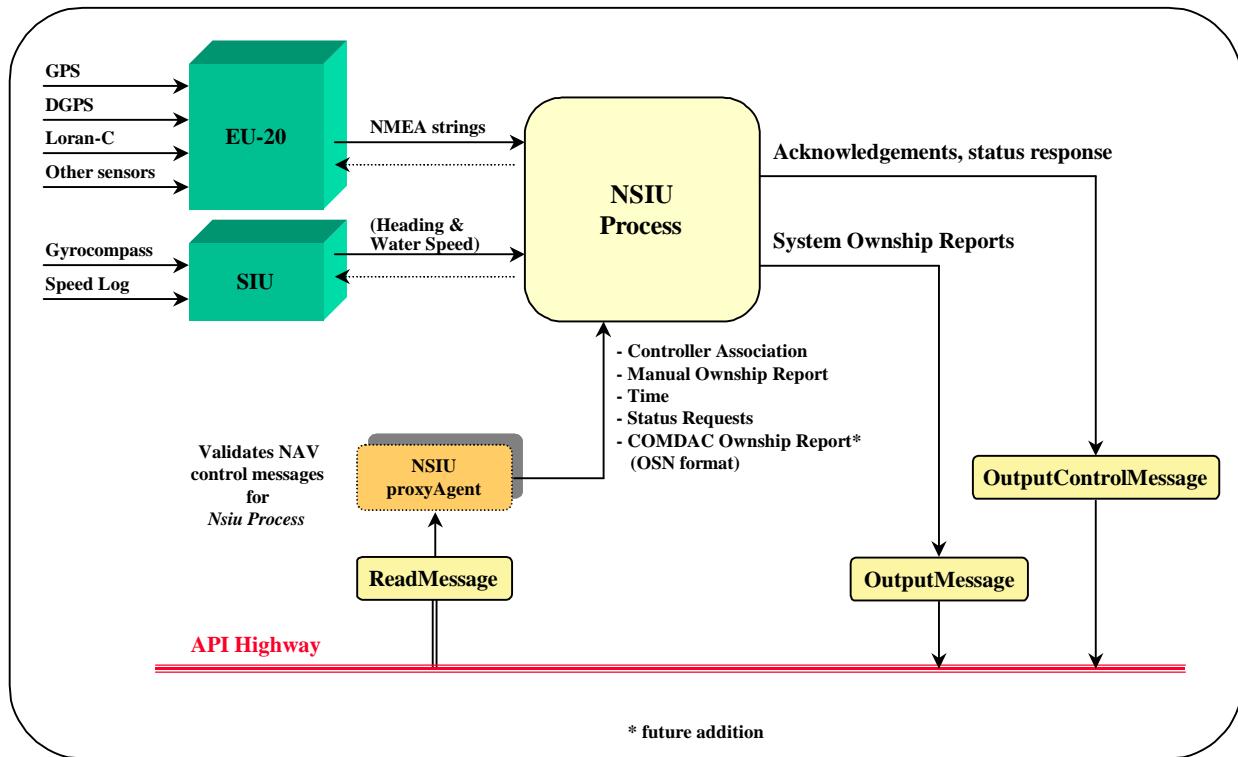


Figure 10. NSIU Functional Model

Only one NSIU process is active at any time, and will be started on the appropriate console. An operator (system master), through InterConnection Control, designates which console is allowed to enter manual navigation data which becomes part of the System Ownship Report. The software at that operator position establishes an association with the available NSIU process. For further detail, see the *InterConnection Control Concept Paper*.

4.5.1 Navigational Sensor Interface

The National Marine Electronics Association standard, NMEA 0183, is a set of codes which are universally accepted as the language of communication between limited-bandwidth marine electronic devices, e.g. GPS, Direction Finder, Depth Sounder, Autopilots, etc. Information is passed as ASCII text divided into "sentences".⁷ A sentence may contain up to 82 characters including the "\$" and CR/LF. NMEA 0183 data are normally sent at 4800 baud.

⁷ Each sentence starts with a "\$", a two letter "talker ID", a three letter "sentence ID", followed by a number of data fields separated by commas, followed by an optional checksum, and terminated by a carriage return/line feed character pair.

4.5.2 NMEA 0183 Interface

The EU-20 NMEA 0183 interface provides 4 talker and up to 12 listener ports. The NSIU process accepts the following sentences:

GGA - Global Positioning System Fix Data

Time, position, and fix related data for a GPS receiver.

GLL - Geographic Position - Latitude/Longitude

Latitude and Longitude of present vessel position, time of position fix and status.

RMC - Recommended Minimum Specific GPS/TRANSIT Data

Time, date, position and speed data provided by a GPS or TRANSIT navigation receiver.

VHW - Water Speed and Heading

The compass heading to which the vessel points and the speed of the vessel relative to the water.

ZDA - Time & Date

UTC, day, month, year and local time zone.

Additionally, Loran C reports and ADF reports can be processed by the NSIU process.

4.5.3 Non-NMEA Navigational Interface

The gyrocompass and speed log inputs consists of analogue data, which are preprocessed by the SIU. The SIU outputs the following NMEA sentences to the NSIU process:

HDM - Heading, Magnetic

Actual vessel heading in degrees Magnetic.

HDT - Heading, True

Actual vessel heading in degrees True produced by any device or system producing true heading.

MWV - Water Speed and Angle

When the Reference Field is set to Relative, data is provided giving the wind angle in relation to the vessel's heading and wind speed, both relative to the (moving) vessel. When the Reference Field is set to True, data is provided giving the wind angle relative to the vessel's heading and wind speed, both with reference to the (moving) water. True wind is the vector sum of the Relative (Apparent) wind vector and the vessel's velocity vector along the heading line of the vessel. It represents the wind at the vessel if it were stationary relative to the water and heading in the same direction.

VBW - Dual Ground/Water Speed

Water referenced and ground referenced speed data.

4.6

Operator Position (OP) Process

The *Operator Position Process*, OP, receives radar image information, system track reports, navigational reports, status messages, and requests from other operator positions. The radar image information describes the format of the requested radar image. The *Radar Input Process*, RI, transforms the image received into a displayable ZPixmap format which is then sent by the *Situation Display Process*, SD, to the X Server.

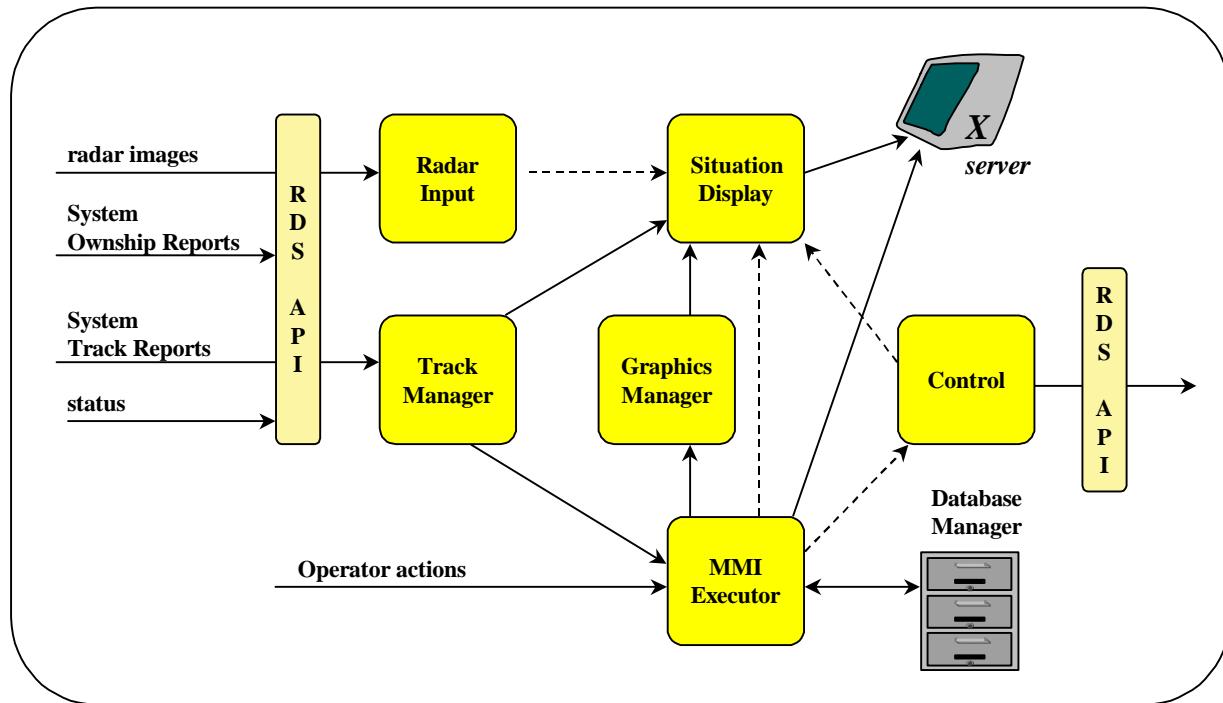


Figure 11. Operator Position Process

The *Track Manager Process* provides display symbology to SD, for the system track reports and also maintains a track file history. The *MMI Executor Process*, MX, handles operator actions and provides menu displays for the operator. MX interprets Tcl/Tk⁸ language statements for the MMI interface. The *Graphics Manager Process*, GM, allows the operator to define polygon and circular regions within the situation display. The *Control Process*, CO, manages BIT status requests/results and heartbeat status from the other AN/SPS-73(V) processes. The *Database Manager Process*, OD, provides access and manipulation of the stored information.

5.

⁸ TCL/TK is a programming system developed by John Ousterhout at the University of California, Berkeley, which is easy to use, and which has very useful graphical interface facilities. TCL is the basic programming language, while TK is a ToolKit of widgets, which are graphical objects similar to those of other GUI toolkits, such as Xlib, Xview and Motif.

RDS Communications Methodology

There are basically three types of users of the RDS COMM software: (1) Service Providers (generators of: tracks, images, navigation reports,...), (2) Controlling Positions (OP & ETS) for these service providers, and (3) Observers (OP & ETS in a view only mode, for radar images and track reports). All users use the ReadMessage function to extract RDS API messages from their input queues. An example communications model is depicted in Figure 12.

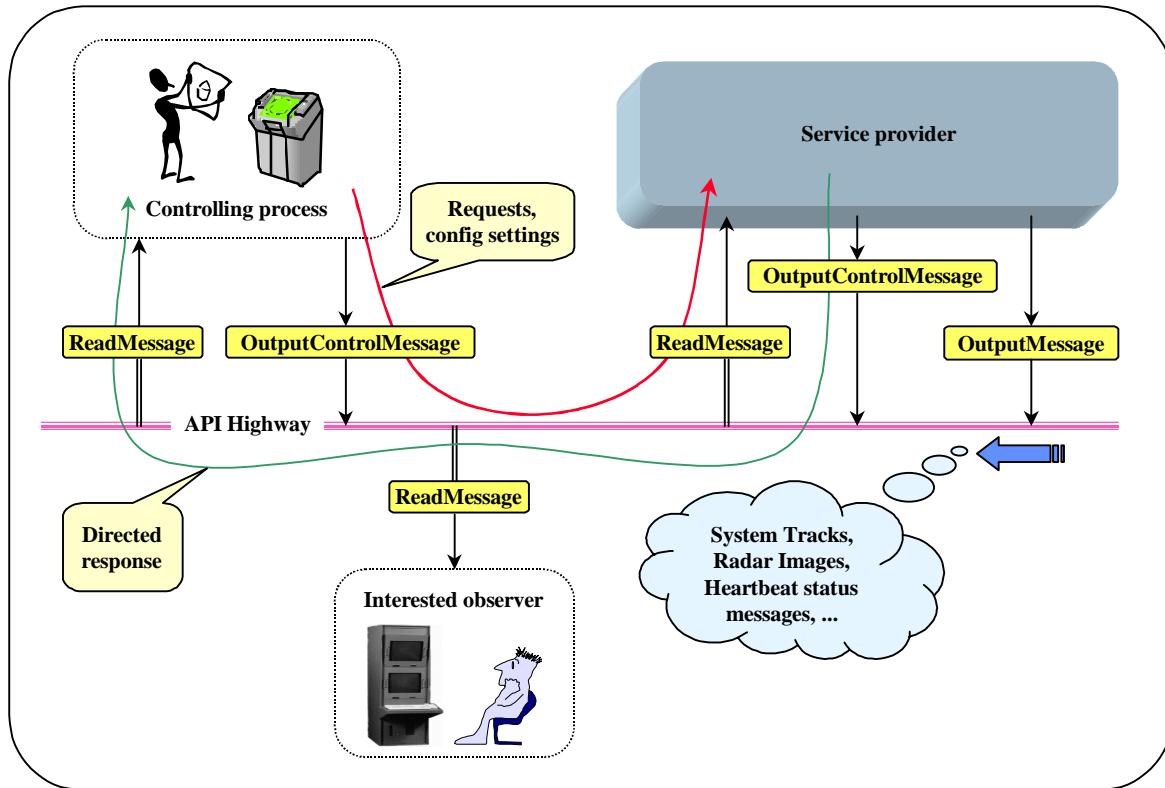


Figure 12. RDS communication model

Service providers typically use the `OutputMessage` function to transmit [broadcast] track reports, ownership reports, and heartbeat status. Along with Radar Images, these messages are sent via UDP packets. Controlling positions (following an association with a service provider) use the `OutputControlMessage` function to direct their requests and configuration settings to the associated service provider. `OutputControlMessage` uses TCP segments for a one-to-one communication interface.

This document only catalogs and describes the content of the RDS API messages. For further details, please refer to the *Radar Data Service Communication User Manual* on how to use the RDS API and see the *InterConnection Control Concept Paper* to find out about establishing associations with service providers and the management of system parameters.

Reference Documents

6.1 Specifications

DTCG23-96-C-ASR009, Contract for Surface Search Radar
26 January 1996

6.2 Standards

DOD-STD-2167A, [Tailored] Defense System Software Development
28 February 1988

6.3 Other Documents, Drawings, and Publications

HSSR-2000, InterConnection Control Concept Paper
25 November 1996

Radar Data Service Communication User Manual, document #TBD
01 July 1997

NMEA 0183, Standard for Interfacing Marine Electronic Devices, Version 2.1

Surface Search Radar Technical Interface Documentation Package, Rev. 2

ANSI X3T9.5, Fiber Distribution Data Interface (FDDI) Station Management

IEEE 802.3, Information Processing Systems, Local Area Networks - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specification of 1993

Hewlett Packard – HP-UX documentation

Wind River Systems – VxWorks documentation

7. RDS API Message Catalog

This section catalogs (via the following tables) the RDS API message sets as related to a particular category. For programming information please refer to the *Radar Data Service Communication User Manual*. Each category contains a list of messages (AMP types), which includes the source of the message, destination, how transmitted, how often, and where to find the data structure definition.

7.1 Category: Any ownship report

The Any Ownship Report category contains navigational data messages derived from external sensor inputs and optionally manual values entered at the NAV Controller's operator position. Table 1, provides a listing of the available message types.

Table 1. CAT_ANY_OWNSHIP_REPORT

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_COMDAC_OWNSHIP_REPORT	ETS	NSIU	UDP	TBD	TBD	TBD	reserved for SCCS message
AMP_MANUAL_OWNSHIP_REPORT	OP	NSIU	TCP	per request	MANUAL_OWNSHIP_REPORT_MSG	nsiu_api.h	use to manually override portion of the system ownship report
AMP_SYSTEM_OWNSHIP_REPORT	NSIU	OP	UDP	4 Hz	SYSTEM_OWNSHIP_REPORT_MSG	nsiu_api.h	ownship navigational and other related data

7.2 Category: Any status

The Any Status category contains various status messages. Table 2, provides a listing of the available message (AMP) types.

Table 2. CAT_ANY_STATUS

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_ANTENNA_INFO					ANTENNA_INFO_MSG		
AMP_GENERIC_RADAR_ORIENTATION	OP	RP	UDP	per request	RP_GENERIC_RADAR_ORIENTATION_MSG	radar_api.h	provide radar image orientation
AMP_RADAR_TIMER_COUNTDOWN	RCC	OP	TCP	1 Hz	RP_TIMER_COUNTDOWN_MSG	radar_api.h	used only during magnetron warm-up
AMP_RCC_STATUS	RCC	OP	TCP	1 Hz	RADAR_STATUS_MSG	radar_api.h	radar and RCC CCA status
AMP_RDP_RIPC_SPO	RP	OP	TCP	once per scan	RP_RIPC_SPO_MSG	radar_api.h	RIPC system performance message
AMP_RDP_RSC_SPO	RP	OP	TCP	once per scan	RP_RSC_SPO_MSG	radar_api.h	RSC system performance message
AMP_RDP_SCAN_STATUS	RP		UDP	once per scan	RP_SCAN_REPORT_MSG	radar_api.h	scan to scan correlation status
AMP_RDP_SPDP_SPO	RP	OP	TCP	once per scan	RP_SPDP_SPO_MSG	radar_api.h	SPDP system performance message
AMP_RDP_STATUS_OUTPUT	RP				RP_STATUS_MSG		

7.3 Category: Any alert

The Any Alert category provides a mechanism that facilitates alert processing. Table 3, provides a listing of the available message (AMP) types.

Table 3. CAT_ANY_ALERT

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_SYSTEM_ALERT		OP	UDP	asynchronous	SYSTEM_ALERT_MSG	alerts_api.h	Contains various alert message text

7.4 Category: Any track data

The Any Track Data category includes track-related messages that originate from RP, Fusion, Operator Position, or from an external track source such as Link-11 tracks. Table 4, provides a listing of the available message (AMP) types.

Table 4. CAT_ANY_TRACK_DATA

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_LINK_11_TRACK_REPORT	ETS	OP	UDP	TBD	TBD	TBD	reserved for external track reports
AMP_SELECT_TRACK_CLASS					TRACK_MESSAGE		
AMP_SENSOR_TRACK_DROP	fusion	RP	UDP	asynchronous	TRACK_DROP_RECORD	tracks_api.h	sensor drop track message
AMP_SENSOR_TRACK_LOST	RP	Fusion	UDP	asynchronous	TRACK_LOST_RECORD	tracks_api.h	sensor lost track message
AMP_SENSOR_TRACK_REPORT	RP	fusion	UDP	once per scan	SENSOR_TRACK_TYPE	tracks_api.h	sensor track message
AMP_SYSTEM_TRACK_REPORT	fusion	OP	UDP	once per scan	SYSTEM_TRACK_RECORD	tracks_api.h	system track message
AMP_TARGET_DETECTION_REPORT	RP		UDP	asynchronous	TARGET_DETECTION_RECORD	tracks_api.h	target detection message
AMP_TARGET_REPORT	RP		UDP	asynchronous	TRACK_TARGET_REPORT	tracks_api.h	target report

7.5

Category: BIT test request

The BIT Test Request category includes BIT selections available to the controlling operator position. Table 5, provides a listing of the available message (AMP) types.

Table 5. CAT_BIT_TEST_REQUEST

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_BIT_GRAPHICS_CARD	OP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic request for graphics board
AMP_BIT_OP_CONSOLE_PORT	OP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic request for console port
AMP_BIT_OP_CPU	OP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic request for daughter board
AMP_BIT_PCMCIA_CARD	OP				IC_DIAGNOSTIC_BIT_REQUEST_MSG		
AMP_BIT_PCMCIA_HARD_DRIVE	OP				IC_DIAGNOSTIC_BIT_REQUEST_MSG		
AMP_BIT_RCC_CARD	OP	RCC	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic request for RCC board
AMP_BIT_RDE_CARD	OP	RP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic request for RDE board
AMP_BIT_RP_CPU	OP	RP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic request for RP CPU
AMP_BIT_RSC_PRIMARY_CARD	OP	RSC	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	OP to request BIT
AMP_BIT_RSC_SECONDARY_CARD	OP	RSC	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	OP to request BIT

7.6 Category: BIT test status

The BIT Test Status category includes BIT results available to the controlling operator position. Table 6, provides a listing of the available message (AMP) types.

Table 6. CAT_BIT_TEST_STATUS

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_BIT_GRAPHICS_CARD	OP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic report for graphics board
AMP_BIT_OP_CONSOLE_PORT	OP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic report for console port
AMP_BIT_OP_CPU	OP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic report for daughter board
AMP_BIT_PCMCIA_CARD		OP			IC_DIAGNOSTIC_BIT_REQUEST_MSG		
AMP_BIT_PCMCIA_HARD_DRIVE		OP			IC_DIAGNOSTIC_BIT_REQUEST_MSG		
AMP_BIT_RCC_CARD	RCC	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic report for RCC board
AMP_BIT_RDE_CARD	RP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic report for RDE board
AMP_BIT_RP_CPU	RP	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	diagnostic report for RP CPU
AMP_BIT_RSC_PRIMARY_CARD	RSC	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	RSC to report status
AMP_BIT_RSC_SECONDARY_CARD	RSC	OP	TCP	per request	IC_DIAGNOSTIC_BIT_REQUEST_MSG	bits_api.h	RSC to report status

7.7 Category: Fusion to RP request

The Fusion to RP Request category defines internal message formats that are used by the Fusion process to modify existing sensor track states.

7.8 Category: Fusion to OP any status

The Fusion to OP Any Status category defines the message formats that are used by the Fusion process. Table 7, provides a listing of the available message (AMP) types.

Table 7. CAT_FUSION_OP_ANY_STATUS

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_SYSTEM_TRACK_DROP	fusion				SYSTEM_TRACK_DROP_RECORD		

7.9 Category: ICC association

The ICC Association category provides a mechanism to allow a designated operator position to establish an association with one or more service providers⁹. Table 8, provides a listing of the available message (AMP) types.

Table 8. CAT_ICC_ASSOCIATION

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_ACKNOWLEDGE		icc	TCP	per request	ICC_ACKNOWLEDGE_MSG	iccontrol_api.h	Sent by an ICC proxy (service provider) in response to an ICC association/identification message
AMP DISSOLVE	ICC	all	TCP	per request	ICC_DISSOLVE_MSG	iccontrol_api.h	Sent by OP to request association be dissolved
AMP_FUSION_IDENTIFICATION	ICC	fusion	TCP	per request	ICC_FUSER_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to establish association with Fusion proxy
AMP_FUSION_DISSOLVE	ICC	fusion	TCP	per request	ICC_FUSER_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to request association be dissolved
AMP_NSIU_IDENTIFICATION	ICC	nsiu	TCP	per request	ICC_NSIU_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to establish association with NSIU proxy

⁹ Service providers (RCC, RP, RSC, Fusion, NSIU tasks) are the source of the various AN/SPS-73 outputs, such as radar images, track reports, and system navigational reports.

AMP_NSIU_DISSOLVE	ICC	nsiu	TCP	per request	ICC_FUSER_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to request association be dissolved
AMP_RADAR_SHUTDOWN	ICC	rcc	TCP	per request	ICC_RADAR_SHUTDOWN_MSG	iccontrol_api.h	Sent by OP to shut off the Furuno radar
AMP_RCC_IDENTIFICATION	ICC	rcc	TCP	per request	ICC_RCC_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to establish association with RCC proxy
AMP_RP_IDENTIFICATION	ICC	ripC	TCP	per request	ICC_RP_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to establish association with RIPC proxy
AMP_RSC_IDENTIFICATION	ICC	rsc	TCP	per request	ICC_RSC_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to establish association with RSC proxy
AMP_SPDP_IDENTIFICATION	ICC	spdp	TCP	per request	ICC_SPDP_ASSOCIATION_MSG	iccontrol_api.h	Sent by OP to establish association with SPDP proxy

7.10 Category: InterConnection Control

The InterConnection Control category defines the message set used for the control of the AN/SPS-73 system configurations. Table 9, provides a listing of the available message (AMP) types.

Table 9. CAT_INTERCONNECTION_CONTROL

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_ASSIGN_LEVEL_ONE_CONTROL	ICC	ICC	TCP	asynchronous	ICC_ASSIGN_LEVEL_ONE_CONTROL_MSG	iccontrol_api.h	assigns primary radar controller
AMP_ASSIGN_LEVEL_TWO_CONTROL	ICC	ICC	TCP	asynchronous	ICC_ASSIGN_LEVEL_TWO_CONTROL_MSG	iccontrol_api.h	assigns secondary image controller
AMP_ASSIGN_NSIU_CONTROL	ICC	ICC	TCP	asynchronous	ICC_ASSIGN_NSIU_CONTROL_MSG	iccontrol_api.h	assigns responsible navigation control
AMP_BP_HEARTBEAT	*	local IC	UDP	periodic	BP_HEARTBEAT_MSG	iccontrol_api.h	backplane service provider heartbeats
AMP_FORWARD_LEVEL_ONE_PARAMS	ICC	ICC	TCP	asynchronous	ICC_FORWARD_LEVEL_ONE_MSG	iccontrol_api.h	
AMP_FORWARD_LEVEL_TWO_PARAMS	ICC	ICC	TCP	asynchronous	ICC_FORWARD_LEVEL_TWO_MSG	iccontrol_api.h	
AMP_FORWARD_RP_PARAMS	ICC	ICC	TCP	asynchronous	ICC_FORWARD_RP_PARAMS_MSG	iccontrol_api.h	
AMP_FORWARD_RP_PARAMS_ACK	ICC	ICC	TCP	asynchronous	ICC_FORWARD_RP_PARAMS_MSG	iccontrol_api.h	
AMP_GENERIC_RADAR_BEARING					ICC_RADAR_BEARING_MSG	iccontrol_api.h	
AMP_ICC_HEARTBEATS	ICC	ICC	UDP	periodic	ICC_HEARTBEAT_MSG	iccontrol_api.h	ICC Broker heartbeats
AMP_ICC_ROLE_ASSIGNMENT_TABLE	ICC	ICC	UDP	asynchronous	ICC_ROLE_ASSIGNMENT_TABLE_MSG	iccontrol_api.h	defines system control designations
AMP_IMAGE_AVAILABILITY	ICC	any OP	UDP	asynchronous	ICC_IMAGE_AVAILABILITY_MSG	iccontrol_api.h	defines available radar images
AMP_LEVEL_ONE_PARAMETERS	ICC	ICC	TCP	asynchronous	ICC_LEVEL_ONE_PARAMETERS_MSG	iccontrol_api.h	current radar parameters
AMP_LEVEL_TWO_PARAMETERS	ICC	ICC	TCP	asynchronous	ICC_LEVEL_TWO_PARAMETERS_MSG	iccontrol_api.h	current scan converter parameters
AMP_PAUSE_RADAR_CONTROL					ICC_SWITCH_CONTROL_MSG	iccontrol_api.h	
AMP_PAUSE_RADAR_ACK					ICC_SWITCH_CONTROL_MSG	iccontrol_api.h	
AMP_RESUME_RADAR_CONTROL					ICC_RESUME_RADAR_CONTROL_MSG	iccontrol_api.h	
AMP_SETUP_DATA_1	ICC	ICC	TCP	asynchronous	ICC_SETUP_DATA_MSG_1	iccontrol_api.h	system configuration data
AMP_SETUP_DATA_2	ICC	ICC	TCP	asynchronous	ICC_SETUP_DATA_MSG_2	iccontrol_api.h	system configuration data
AMP_SWITCH_RADARS	ICC	ICC	TCP	asynchronous	ICC_SWITCH_CONTROL_MSG	iccontrol_api.h	
AMP_SWITCH_RADAR_ACK	ICC	ICC	TCP	asynchronous	ICC_SWITCH_CONTROL_MSG	iccontrol_api.h	

7.11 Category: Network control

The Network Control category defines internal message formats that are used by the RDS software.

7.12 Category: NSIU any status

The NSIU Any Status category provides NSIU operational status for the controlling NAV operator position. Table 10, provides a listing of the available message (AMP) types.

Table 10. CAT_NSIU_ANY_STATUS

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_ADF_MESSAGE		OP	TCP	per request	ADF_RECORD	nsiu_api.h	contain VHF contact information (bearing & frequency)
AMP_SIU_STATUS_CONTROL_REQUEST	OP	NSIU	TCP	per request	SIU_STATUS_CONTROL_MSG	nsiu_api.h	
AMP_SIU_STATUS_CONTROL_RESPONSE	NSIU	OP	TCP	per request	SIU_STATUS_CONTROL_MSG	nsiu_api.h	
AMP_NAV_SENSOR_STATUS_REQUEST	OP	NSIU	TCP	per request	NAV_SENSOR_STATUS_MSG	nsiu_api.h	
AMP_NAV_SENSOR_STATUS_RESPONSE	NSIU	OP	TCP	per request	NAV_SENSOR_STATUS_MSG	nsiu_api.h	

7.13 Category: OP to Fusion operator entry

The OP to Fusion Operator Entry category provides a mechanism for operator positions to control the system track states. Table 11, provides a listing of the available message (AMP) types.

Table 11. CAT_OP_FUSION_OPERATOR_ENTRY

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_EXECUTE_DEFUSE_SYSTEM_TRACK	OP	fusion	UDP	per request	SYSTEM_TRACK_DEFUSE_REQUEST	tracks_api.h	system track defuse request
AMP_EXECUTE_DROP_SYSTEM_TRACK	OP	fusion	UDP	per request	SYSTEM_TRACK_DROP_RECORD	tracks_api.h	drop system track message
AMP_EXECUTE_FUSE_SYSTEM_TRACK	OP	fusion	UDP	per request	SYSTEM_TRACK_FUSE_REQUEST	tracks_api.h	Fuse two system tracks request
AMP_SELECT_SYSTEM_TRACK_CLASS	OP	fusion	UDP	per request	SYSTEM_TRACK_CLASS_SELECT_MSG	tracks_api.h	system track class request message
AMP_SELECT_SYSTEM_TRACK_AFFIL	OP	fusion	UDP	per request	SYSTEM_TRACK_AFFIL_SET_MSG	tracks_api.h	system track class request message
AMP_SET_SYSTEM_TRACK_LABEL	OP	fusion	UDP	per request	SYSTEM_TRACK_LABEL_SET_MSG	tracks_api.h	set system track label message
AMP_SET_SYSTEM_TRACK_BEAM	OP	fusion	UDP	per request	SYSTEM_TRACK_BEAM_SET_MSG	tracks_api.h	
AMP_SET_SYSTEM_TRACK_LENGTH	OP	fusion	UDP	per request	SYSTEM_TRACK_LENGTH_SET_MSG	tracks_api.h	
AMP_SET_FUSION_CORR_GATE	OP	fusion	UDP	per request	FUSION_SET_CORR_GATE_MSG	tracks_api.h	
AMP_SELECT_AUTO_FUSION_TRACK	OP	fusion	UDP	per request	FUSION_SELECT_AUTO_MSG	tracks_api.h	
AMP_SELECT_SYSTEM_TRACK_BOARD	OP	fusion	UDP	per request	SYSTEM_TRACK_BOARD_SET_MSG	tracks_api.h	

7.14 Category: OP graphic definition

The OP Graphic Definition category defines messages that specify a particular graphic area, to be used in the manipulation of detected tracks. Table 12, provides a listing of the available message (AMP) types.

Table 12. CAT_OP_GRAPHIC_DEFINITION

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_GRAPHIC_CIRCLE	OP	RP	TCP	per request	GRAPHIC_MESSAGE	graphic_api.h	inform of operator circle definition
AMP_GRAPHIC_COMMAND	OP	RP	TCP	per request	GRAPHIC_COMMAND_RECORD	graphic_api.h	drop, update, activate graphics
AMP_GRAPHIC_POLYLINE	OP	RP	TCP	per request	GRAPHIC_MESSAGE	graphic_api.h	inform of operator circle definition
AMP_GRAPHIC_SECTOR	OP	RP	TCP	per request	GRAPHIC_MESSAGE	graphic_api.h	

7.15 Category: OP to NSIU operator entry

The OP to NSIU Operator Entry category allows the controlling NAV operator position to define a time setting for the NSIU. Table 13, provides a listing of the available message (AMP) types.

Table 13. CAT_OP_NSIU_OPERATOR_ENTRY

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_EXECUTE_ADF_RETRIEVAL	OP	NSIU	TCP	per request	NSIU_MESSAGE	nsiu_api.h	
AMP_SET_SYSTEM_TIME	OP	NSIU	TCP	per request	SYSTEM_TIME_SET_RECORD	nsiu_api.h	

7.16

Category: OP to RCC operator entry

The OP to RCC Operator Entry category defines messages for control of the attached radars. Table 14, provides a listing of the available message (AMP) types.

Table 14. CAT_OP_RCC_OPERATOR_ENTRY

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_SELECT_FILTERING	OP	RCC	TCP	per request	FILTER_SELECT_MSG	radar_api.h	enable/disable RCC low pass filters
AMP_SELECT_PM	OP	RCC	TCP	per request	PERFORMANCE_MONITOR_SELECT_MSG	radar_api.h	enable/disable Furuno performance monitor
AMP_SELECT_RADAR_STATE	OP	RCC	TCP	per request	RP_STATE_SELECT_MSG	radar_api.h	command radar to standby, radiate, power off
AMP_SELECT_VIDEO_SOURCE					RCC_VIDEO_SELECTION_MSG		
AMP_SET_AUTO_RF_TUNE	OP	RCC	TCP	per request	RF_TUNING_SELECT_MSG	radar_api.h	enable SSR auto tune of Furuno video
AMP_SET_BLANKING_ZONE	OP	RCC	TCP	per request	ICC_MSG_RCVR_MAP		
AMP_SET_MAIN_BANG					MAIN_BANG_CONTROL_MSG		
AMP_SET_MANUAL_RF_TUNE	OP	RCC	TCP	per request	RF_TUNING_CONTROL_MSG	radar_api.h	set coarse/fine tune controls for Furuno input video
AMP_SET_RCC_PARAMETERS	OP	RCC	TCP	per request	RCC_PARAMETERS_MSG	radar_api.h	define radar settings

7.17

Category: OP to RDE operator entry

The OP to RDE Operator Entry category settings for the Radar Data Extractor card. Table 15, provides a listing of the available message (AMP) types.

Table 15 CAT_OP_RDE_OPERATOR_ENTRY

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_SET_GUARD_CELLS	OP	RP	TCP	per request	RP_GUARD_CELLS_SET_MSG	radar_api.h	set guard cell message
AMP_SET_SCALER_VALUE	OP	RP	TCP	per request	RP_RVF_SCALER_SET_MSG		
AMP_SET_RANGE_BIAS	OP	RP	TCP	per request	RP_RANGE_BIAS_SET_MSG		
AMP_SET_TBM_BIASES	OP	RP	TCP	per request	RP_TBM_BIASES_SET_MSG	radar_api.h	threshold bias map message
AMP_SET_TRACK_SCAN_COUNT	OP	RP	TCP	per request	TRACK_SCAN_COUNT_SET_MSG	tracks_api.h	set track scan count message
AMP_SELECT_NEAR_LAND	OP	RP	TCP	per request	NEAR_LAND_MODE_SELECT_MSG		

7.18 Category: OP to RP operator entry

The OP to RP Operator Entry category defines the message set for controlling the RP from a level-1 operator position. Table 16, provides a listing of the available message (AMP) types.

Table 16. CAT_OP_RP_OPERATOR_ENTRY

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_EXECUTE_MANUAL_SENSOR_TRACK_ACQUIRE	OP	RP	TCP	per request	TRACK_ACQUIRE_REQUEST	tracks_api.h	request the RP to acquire a track
AMP_EXECUTE_RESET	OP	RP	TCP	per request	RP_RESET_EXECUTE_MSG	radar_api.h	reset parameters
AMP_LAND_MASK_MAP_DOWNLOAD					LAND_MASK_DOWNLOAD_RECORD		
AMP_LAND_MASK_SELECT					LAND_MASK_SELECT_RECORD		
AMP_SELECT_AGC_TOGGLE	OP	RP	TCP	per request	RP_AGC_SELECT_MSG	radar_api.h	AGC on/off selection
AMP_SELECT_AUTO_MODE	OP	RP	TCP	per request	RP_AUTO_MODE_SELECT_MSG	radar_api.h	select auto mode parameters
AMP_SELECT_ENUM_RANGE_SCALE					ENUM_RANGE_SCALE_SELECT_MSG		
AMP_SELECT_IR_RISE					IR_RISE_SELECT_MSG		
AMP_SELECT_IR_FALL					IR_FALL_SELECT_MSG		
AMP_SELECT_MANUAL_MODE	OP	RP	TCP	per request	RP_MANUAL_MODE_SELECT_MSG	radar_api.h	select manual mode parameters
AMP_SELECT_PULSE_IR	OP	RP	TCP	per request	RP_PULSE_IR_SELECT_MSG	radar_api.h	set interference rejection on/off
AMP_SELECT_RANGE_SCALE	OP	RP	TCP	per request	RP_RANGE_SCALE_SELECT_MSG	radar_api.h	range selection message
AMP_SELECT_STC_TOGGLE	OP	RP	TCP	per request	RP_STC_SELECT_MSG	radar_api.h	sensitivity time constant toggle message
AMP_SET_ACP_THRESHOLD	OP	RP	TCP	per request	RP_ACP_THRESHOLD_SET_MSG	radar_api.h	set ACP threshold
AMP_SET_ARP_THRESHOLD	OP	RP	TCP	per request	RP_ARP_THRESHOLD_SET_MSG	radar_api.h	set ARP threshold
AMP_SET_BIAS_CORRECTION	OP	RP	TCP	per request	RP_AZIMUTH_BIAS_SET_MSG	radar_api.h	set bias correction message
AMP_SET_CONDITION					RP_CONDITION_MODE_SELECT_MSG		
AMP_SET_FTC	OP	RP	TCP	per request	RP_FTC_SET_MSG	radar_api.h	set fast time constant message
AMP_SET_RP_PARAMETERS	OP	RP	TCP	per request	RP_PARAMETERS_MSG	radar_api.h	set radar parameters selection
AMP_SET_SENSOR_TRACK_PRIORITY_AREA					SENSOR_TRACK_PRIORITY_AREA_SET_MSG		
AMP_SET_SENSOR_TRACK_SPEEDS	OP	RP	TCP	per request	TRACK_SPEED_FILTER_SET_MSG	tracks_api.h	set sensor track speed
AMP_SET_STC_CURVE	OP	RP	TCP	per request	RP_STC_SET_CURVE_MSG	radar_api.h	set sensitivity time constant curve definition
AMP_SET_TRIG_THRESHOLD	OP	RP	TCP	per request	RP_TRIG_THRESHOLD_SET_MSG	radar_api.h	trigger threshold and delay
AMP_SET_VIDEO	OP	RP	TCP	per request	RP_VIDEO_SET_MSG	radar_api.h	set video bias and gain

7.19

Category: OP to RSC operator entry

The OP to RSC Operator Entry category defines a message set for the control of a Radar Scan Converter process by a controlling operator position., Table 17 provides a listing of the available message (AMP) types.

Table 17. CAT_OP_RSC_OPERATOR_ENTRY

AMP Type	Source	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_RADAR_IMAGE_PRESENTATION	OP	RSC	TCP	per request	RP_IMAGE_PRESENTATION_SELECT_MSG	radar_api.h	radar image in X_Y or RHO_THETA format
AMP_SELECT_LANDMASK_VIEW	OP	RSC	TCP	per request	RP_LAND_MASK_SELECT_MSG	radar_api.h	show landmask verse radar image
AMP_SET_DECAY_RATE	OP	RSC	TCP	per request	RP_DECAY_RATE_SET_MSG	radar_api.h	set decay rate for image
AMP_SET_ORIGIN_OFFSET	OP	RSC	TCP	per request	RP_ORIGIN_OFFSET_SET_MSG	radar_api.h	offset for radar image
AMP_SET_THRESHOLD_FORMAT	OP	RSC	TCP	per request	RP_THRESHOLD_FORMAT_SET_MSG	radar_api.h	threshold of detection
AMP_SET_RADAR_IMAGE	OP	RSC	TCP	per request	RADAR_IMAGE_SELECT_MSG		
AMP_SELECT_ORIENTATION	OP	RSC	TCP	per request	RP_SELECT_ORIENTATION_MSG	radar_api.h	in head_up, north_up, course_up, true motion
AMP_SET_RSC_PARAMETERS	OP	RSC	TCP	per request	RSC_PARAMETERS_MSG		

7.20 Category: Radar image data

The Radar Image Data category includes primary and secondary radar images that could be available within the AN/SPS-73 system based on the number of RPs and attached radar/transmitter units. Table 18, provides a listing of the available message (AMP) types.

Table 18 : CAT_RADAR_x_ANY_IMAGE (x = 1 to 4)

AMP Type	Source.	Dest.	Msg. Level	Update Frequency	Data Structure	Header File	Description
AMP_PRIMARY_IMAGE	1 st RSC	any OP	TCP	once per scan	RP_RADAR_IMAGE_MSG	radar_api.h	primary scan converter output
AMP_SECONDARY_IMAGE	2 nd RSC	any OP	TCP	once per scan	RP_RADAR_IMAGE_MSG	radar_api.h	secondary scan converter output

API Message Formats

The following sections illustrate the contents of the RDS API Messages; every API message starts with a header structure as depicted in Table 19 and defined in the `messages_api.h` file. Each table includes a description of the message field, its data type, unit of measure, range, and its precision.

Table 19. API message Header structure

Message Field	Description	Data Type	Units	Range	Precision
type	Object Type (internal usage only)	unsigned int :8	N/A	bit field	N/A
len	Object Word Length (internal usage only)	unsigned int :8	N/A	bit field	N/A
subtype	message “CAT” (category) type	unsigned int :8	enum	bit field	N/A
amptype	message “AMP” type	unsigned int :8	enum	bit field	N/A
count	message size	unsigned int :32	bytes	radar image size	N/A
source_node	message originator: IP address	unsigned int	N/A		N/A
process_id	message originator: VxWorks Task ID or HP-UX Process ID	unsigned int	cardinal	N/A	N/A
timeStamp	message transmission time stamp	struct timeval (sys/time.h)	seconds	finite	micro-seconds

*NOTE: Always refer to the current RDS Comm <install_dir>/include directory
for the most up-to-date version of the API message content.*

8.1

Radar Image Message (primary & secondary format)

Table 20. AMP_any_IMAGE (where any can be {PRIMARY | SECONDARY})

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
lat	Latitude of radar image center	double	degrees		
lon	Longitude of radar image center	double	degrees		
position_status	lat/lon position data status indicator	STATUS_TYPE	enum	INVALID, VALID, PENDING	N/A
time	seconds since midnight 1 January, 1970	struct timeval	seconds	finite	1 second
x_offset	from center of screen	signed int	meters	0 - 177792	1 meter
y_offset	from center of screen	signed int	meters	0 - 177792	1 meter
rotation	relative to the Bow	float	degrees	0.0 - 359.9	0.1 degree
size	radar image size	long	bytes	0 - 512KB	N/A
range_scale		long	meters	0 - 177792	
orientation		RP_RADAR_ORIENTATION	enum	ORIENTATION_HEAD, ORIENTATION_NORTH, ORIENTATION_COURSE, ORIENTATION_GYRO_COURSE,	N/A
compression	indicates if the radar image is compressed (i.e., RLE)	long	enum	0 = Disable, 1 = Enable	N/A
image_format	XY format or Rho Theta	RP_RADAR_IMAGE_TYPE	enum	X_Y_FORMAT, RHO_THETA_FORMAT	N/A
pixels_per_byte	number of pixels per byte (always 2)	long	cardinal	2	N/A
antennae_scan_time	antenna rotation time	float	seconds	0.9 - 12.0	.1 second
image_bits	radar image density, 2 or 4 bits	RP_RADAR_IMAGE_BITS	enum	IMAGE_2_BIT, IMAGE_4_BIT	N/A
radar_image_data[]	radar image	unsigned char	bytes	512KB	N/A

8.2

API message listings by AMP type.

The remaining API messages are listed in alphabetical order.

Table 21. AMP_ACKNOWLEDGE (ICC_ACKNOWLEDGE_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
servProv	Identity of originating task	ICC_NODE	enum	NODE_ICC_MASTER, NODE_ICC_CLIENT, NODE_OP, NODE_RP, NODE_SPDP, NODE_RIPC_1, NODE_RIPC_2, NODE_RSC, NODE_RCC, NODE_NSU, NODE_FUSION	N/A
imageSelection	Indication of affected image	ICC_RSC_IMAGE	enum	PRIMARY, SECONDARY	N/A

Table 22. AMP_ADF_MESSAGE (ADF_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
messagetype	Reserved space to identify message type	HEADER	N/A	N/A	N/A
bearing	bearing of transmitter from OS	float	degrees	0 - 359.99	.01 degree
type	bearing type	DIRECTION_TYPE	enum	M, T	N/A
time	Contact message timestamp (unix seconds)	unsigned long	seconds	0 - 4294967295	1 second
pad	To allow alignment (size of struct) w/ HPUX compiler	unsigned long	N/A	N/A	N/A
freq	VHF frequency	double	MHz	0 - 99.999	.001 MHz

Table 23. AMP_ASSIGN_LEVEL_ONE_CONTROL (ICC_ASSIGN_LEVEL_ONE_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
rpStation	Name of console containing radar processing capability	STATIONS	enum	ALOFT_CONN, BRIDGE_NAV, BRIDGE_PORT, BRIDGE_SHIPPING, BRIDGE_STBD, CIC_NAV, CIC_SHIPPING, SCCS_1, SCCS_2, SCCS_3, SCCS_4, JMCIS	N/A
jBoxSwitched	Indicates current J-Box setting	BOOL	enum	FALSE, TRUE	N/A
selectedRadarPort	Indicates currently selected radar video	PORT_SELECTION	enum	FURUNO_PORT, GENERIC_PORT_1, GENERIC_PORT_2	N/A
furunoParams	Parameters for Furuno radar being controlled, see header file for definition of record	RADAR_PARAMETERS	N/A	N/A	N/A
genericOneParams	Parameters for radar video on generic port 1, see header file for definition of record	RADAR_PARAMETERS	N/A	N/A	N/A
genericTwoParams	Parameters for radar video on generic port 2, see header file for definition of record	RADAR_PARAMETERS	N/A	N/A	N/A

Table 24. AMP_ASSIGN_LEVEL_TWO_CONTROL (ICC_ASSIGN_LEVEL_TWO_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
rpStation	Name of console containing radar processing capability	STATIONS	enum	ALOFT_CONN, BRIDGE_NAV, BRIDGE_PORT, BRIDGE_SHIPPING, BRIDGE_STBD, CIC_NAV, CIC_SHIPPING, SCCS_1, SCCS_2, SCCS_3, SCCS_4, JMCIS	N/A
jBoxSwitches	Indicates current J-Box setting	BOOL	enum	FALSE, TRUE	N/A
selectedRadarPort	Indicates currently selected radar video	PORT_SELECTION	enum	FURUNO_PORT, GENERIC_PORT_1, GENERIC_PORT_2	N/A
furunoParams	Parameters for Furuno radar being controlled, see header file for definition of record	RSC_PARAMETERS	N/A	N/A	N/A
genericOneParams	Parameters for radar video on generic port 1, see header file for definition of record	RSC_PARAMETERS	N/A	N/A	N/A
genericTwoParams	Parameters for radar video on generic port 2, see header file for definition of record	RSC_PARAMETERS	N/A	N/A	N/A

Table 25. AMP_ASSIGN_NSIU_CONTROL (ICC_ASSIGN_NSIU_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A

Table 26. AMP_BP_HEARTBEAT (BP_HEARTBEAT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
count	Running counter	unsigned long	cardinal	finite	1 message
type	node type	ICC_NODE	N/A	NODE_ICC_MASTER, NODE_ICC_CLIENT, NODE_OP, NODE_RP, NODE_SPDP, NODE_RIPC_1, NODE_RIPC_2, NODE_RSC, NODE_RCC, NODE_NSIU, NODE_FUSION	N/A
status	node status	unsigned long	bit mask	TBD	N/A

Table 27. AMP_DISSOLVE (ICC_DISSOLVE_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
imageSelection	indication of affected image	ICC_RSC_IMAGE	enum	PRIMARY, SECONDARY	N/A

Table 28. AMP_EXECUTE_DEFUSE_SYSTEM_TRACK (SYSTEM_TRACK_DEFUSE_REQUEST)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify msg. type	HEADER	N/A	N/A	N/A
time	Time stamp (in seconds since midnight)	double	seconds	finite	1 microsecond
cti	Sys. track number (1-1000)	unsigned short	cardinal	1 - 1000	N/A
tsi	Fusion source identification	unsigned int	cardinal	1 - 255	N/A

Table 29. AMP_EXECUTE_DROP_SYSTEM_TRACK (SYSTEM_TRACK_DROP_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
cti	Contact Track Identification	unsigned short	cardinal	1 - 1000	N/A
tsi	Track Source Identification	unsigned int	cardinal	1 - 255	N/A
systemTrackStatus	Reason dropped	DROP_SYS_STATUS	enum	LOST_SYSTRK, DROP_SYSTRK, DROP_SYSTRK_NOTF	N/A

Table 30. AMP_EXECUTE_FUSE_SYSTEM_TRACK (SYSTEM_TRACK_FUSE_REQUEST)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message type	HEADER	N/A	N/A	N/A
time	Time stamp (in seconds since 1970)	double	seconds	finite	1 microsecond
tsi	Fusion source identification	unsigned int	cardinal	1 - 255	N/A
primary_track	System track number	unsigned short	cardinal	1 - 1000	N/A
secondary_track	System track number	unsigned short	cardinal	1 - 1000	N/A

Table 31. AMP_EXECUTE_MANUAL_SENSOR_TRACK_ACQUIRE (TRACK_ACQUIRE_REQUEST)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
x_pos	Acquire X position (meters)	long	TBD	TBD	TBD
y_pos	Acquire Y position (meters)	long	TBD	TBD	TBD
vts_cti	VTS label	long	TBD	TBD	TBD
duration	In seconds, if 0 perform acquire if non zero attempt. Acquire for number of seconds specified	float	TBD	TBD	TBD
search_radius	Search radius for request (meters)	unsigned long	TBD	TBD	TBD

Table 32. AMP_EXECUTE_RESET (RP_RESET_EXECUTE_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
delay	Delay before reset (in seconds)	unsigned long	seconds	finite	1 second
restart_mode	cold restart: go back to factory defaults warm restart: restore last saved parameters	RP_RESTART_TYPE	enum	COLD_RESTART, WARM_RESTART	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 33. AMP_FUSION DISSOLVE (ICC_FUSER DISSOLVE_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A

Table 34. AMP_FUSION IDENTIFICATION (ICC_FUSER ASSOCIATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
Fusion_ID	site ID	unsigned int	cardinal	1 - 255	N/A

Table 35. AMP_GENERIC_RADAR_ORIENTATION (RP_GENERIC_RADAR_ORIENTATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
radar_type	the type of radar	RADARS	enum	X_BAND, S_BAND, X_BAND_BOW, X_BAND_STERN, X_BAND_PORT, X_BAND_STBD, MK_92, SPS_67, SPS_55	N/A
azimuth_orientation	the orientation of the RP azimuth	RP_AZIMUTH_ORIENTATION	enum	HEADING_RELATIVE, NORTH_RELATIVE	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 36. AMP_GRAPHIC_CIRCLE (GRAPHIC_MESSAGE)

Message Field	Description	Data Type	Units	Range	Precision
header	Header to identify messages	HEADER	N/A	N/A	N/A
circ	Circle definition message, see header file for definition of record	CIRCLE_RECORD	N/A	N/A	N/A
poly	Polyline definition message, see header file for definition of record	POLYLINE_RECORD	N/A	N/A	N/A
sect	Sector definition message, see header file for definition of record	SECTOR_RECORD	N/A	N/A	N/A
cmd	Graphic Enable/Disable command, see header file for definition of record	GRAPHIC_COMMAND_RECORD	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 37. AMP_GRAPHIC_COMMAND (GRAPHIC_COMMAND_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
gid	Graphic ID: 0=All; 1-64K=Valid	unsigned short	N/A	0 - 64	1
pad	To allow alignment with HPUX compiler	unsigned short	N/A	N/A	N/A
type	Graphic Type	GRAPHIC_TYPE	enum	GRAPHIC_TYPE_NONE, GRAPHIC_TYPE_LAND_MASK, GRAPHIC_TYPE_RADAR_LAND MASK, GRAPHIC_TYPE_ACQUISITION_ WINDOW, GRAPHIC_TYPE_GUARD_ALARM ,, GRAPHIC_TYPE_HIGH_INTEREST ,, GRAPHIC_TYPE_BLANKING_ZON E, GRAPHIC_TYPE_THRESHOLD_BI AS_MAP	N/A
command	Graphic Command	GRAPHIC_COMMAND	enum	GRAPHIC_ADD_UPDATE, GRAPHIC_DROP, GRAPHIC_ACTIVATE, GRAPHIC_DEACTIVATE	N/A

Table 38. AMP_GRAPHIC_POLYLINE (GRAPHIC_MESSAGE)

Message Field	Description	Data Type	Units	Range	Precision
header	Header to identify messages	HEADER	N/A	N/A	N/A
circ	Circle definition message, see header file for definition of record	CIRCLE_RECORD	N/A	N/A	N/A
poly	Polyline definition message, see header file for definition of record	POLYLINE_RECORD	N/A	N/A	N/A
sect	Sector definition message, see header file for definition of record	SECTOR_RECORD	N/A	N/A	N/A
cmd	Graphic Enable/Disable command, see header file for definition of record	GRAPHIC_COMMAND_RECORD	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 -7	1

Table 39. AMP_ICC_HEARTBEATS (ICC_HEARTBEAT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	Reserved space to identify message type	HEADER	N/A	N/A	N/A
count	Running counter	unsigned long	cardinal	finite	1 message
proxyStatus	service provider status	unsigned long	bit mask	TBD	N/A
rolesOnThisOP	assumed roles	ICC_ROLES	enum	bit mask	N/A
consoleID	System node ID	STATIONS	enum	ALOFT_CONN, BRIDGE_NAV, BRIDGE_PORT, BRIDGE_SHIPPING, BRIDGE_STBD, CIC_NAV, CIC_SHIPPING, SCCS_1, SCCS_2, SCCS_3, SCCS_4, JMCIS	N/A
partNumber	component identifier	PART_NUMBER	enum	SAOP_1720549100, SAOP_1720549104, SAOP_1720549107, SAOP_1720549108, SAOP_1720549111, TBOP_172048100, TBOP_1720548102, RP_1720570104,	N/A

Table 40. AMP_IMAGE_AVAILABILITY (ICC_IMAGE_AVAILABILITY_MSG)

Message Field	Description	Data Type	Units	Range	Precision
msgHeader	identifies message format	HEADER	N/A	N/A	N/A
radarProcessor []	defines available radar images per RP	ICC_IMAGE_SELECTION	N/A	MAX_RP_UNITS	N/A

Table 41. AMP_LAND_MASK_MAP_DOWNLOAD (LAND_MASK_DOWNLOAD_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
acp_count	Specified current azimuth heading	unsigned short	ACP	0 - 4095	1 ACP
crossover_count	Specified number of range crossovers for the current azimuth (1-MAX_RANGE_CROSSEOVERS)	unsigned short	cardinal	1 - MAX_RANGE_CROSSEOVERS	N/A
range_crossover [MAX_RANGE_CROSSEOVERS]	Range crossover parameters	RANGE_CROSSOVER	TBD	TBD	TBD

Table 42. AMP_LAND_MASK_SELECT (LAND_MASK_SELECT_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
geo_id	Graphic ID	GEO_ID	enum	GEO_ID_LAND, GEO_ID_ANCHORAGE, GEO_ID_WATER_AREA, GEO_ID_AUTO_ACQUIRE_1, GEO_ID_AUTO_ACQUIRE_2, GEO_ID_AUTO_ACQUIRE_3, GEO_ID_AUTO_ACQUIRE_4, GEO_ID_AUTO_ACQUIRE_5, GEO_ID_AUTO_ACQUIRE_6, GEO_ID_AUTO_ACQUIRE_7, GEO_ID_AUTO_ACQUIRE_8	N/A
map_state	Land Mask Map State	unsigned char	enum	ENABLE, DISABLE	N/A
pad	To allow alignment (size of struct) with HPUX compiler	unsigned short	N/A	N/A	N/A

Table 43. AMP_LEVEL_ONE_PARAMETERS (ICC_LEVEL_ONE_PARAMETERS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
radarNumber	number of radar to which parameters apply	POSSIBLE_RADAR_NUMS	enum	RADAR_ONE, RADAR_TWO, RADAR_THREE, RADAR_FOUR, RADAR_FIVE, RADAR_SIX,	N/A
radarParams	radar's parameter values, see header file for definition of record	RADAR_PARAMETERS	N/A	N/A	N/A

Table 44. AMP_LEVEL_TWO_PARAMETERS (ICC_LEVEL_TWO_PARAMETERS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
radarNumber	number of radar to which parameters apply	POSSIBLE_RADAR_NUMS	enum	RADAR_ONE, RADAR_TWO, RADAR_THREE, RADAR_FOUR, RADAR_FIVE, RADAR_SIX,	N/A
radarParams	radar's parameter values, see header file for definition of record	RSC_PARAMETERS	N/A	N/A	N/A

Table 45. AMP_MANUAL_OWNSHIP_REPORT (MANUAL_OWNSHIP_REPORT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Msg header	HEADER	N/A	N/A	N/A
report_source	Report source identification	int	N/A	-2147483648 - 2147483647	N/A
label [LABEL_SIZE]	Ship's name	char	ASCII	1 - 15	N/A
nav_data	Ship's NAV data information, see header file for definition of record	NAV_DATA_RECORD	N/A	N/A	N/A
nav_current_source	Ship's NAV Cur. source info., see header file for definition of record	NAV_SOURCE_RECORD	N/A	N/A	N/A
nav_alternate_source	Ship's NAV Alt. source info., see header file for definition of record	NAV_SOURCE_RECORD	N/A	N/A	N/A

Table 46. AMP_NAV_SENSOR_STATUS_REQUEST (NAV_SENSOR_STATUS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Msg header	HEADER	N/A	N/A	N/A
nav_sensor_status_device	Device for status	NAV_SENSOR_STATUS_DEVICE_TYPE	enum	DEVICE_DGPS, DEVICE_GPS, DEVICE_RAYNAV, DEVICE_SATNAV, DEVICE_SPEEDLOG, DEVICE_GYRO	N/A
nav_sensor_status_msg_cnt	Number of lines in following msg array	long	count	0 - 4	N/A
nav_sensor_status_msgs [4] [28]		char	N/A	N/A	N/A

Table 47. AMP_NAV_SENSOR_STATUS_RESPONSE (NAV_SENSOR_STATUS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Msg header	HEADER	N/A	N/A	N/A
nav_sensor_status_device	Device for status	NAV_SENSOR_STATUS_DEVICE_TYPE	enum	DEVICE_DGPS, DEVICE_GPS, DEVICE_RAYNAV, DEVICE_SATNAV, DEVICE_SPEEDLOG, DEVICE_GYRO	N/A
nav_sensor_status_msg_cnt	Number of lines in following msg array	long	count	0 - 4	N/A
nav_sensor_status_msgs [4] [28]		char	N/A	N/A	N/A

Table 48. AMP_NSIU DISSOLVE (ICC_NSIU DISSOLVE_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A

Table 49. AMP_NSIU IDENTIFICATION (ICC_NSIU ASSOCIATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
setupParams	navigation parameters, see header file for definition of record	NAV_PARAMETERS	N/A	N/A	N/A

Table 50. AMP_PAUSE_RADAR_CONTROL (ICC_RADAR_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A

Table 51. AMP_RADAR_IMAGE_PRESENTATION (RP_IMAGE_PRESENTATION_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
format	Radar image presentation selection	RP_PRESENTATION_TYPE	enum	X_Y_FORMAT, RHO_THETA_FORMAT	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 -7	1

Table 52. AMP_RADAR_SHUTDOWN (ICC_RADAR_SHUTDOWN_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A

Table 53. AMP_RADAR_TIMER_COUNTDOWN (RP_TIMER_COUNTDOWN_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
timer_countdown	seconds remaining in warmup	unsigned long			
periodic	in Hz	unsigned long			

Table 54. AMP_RCC_IDENTIFICATION (ICC_RCC_ASSOCIATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A

Table 55. AMP_RCC_STATUS (RADAR_STATUS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
ordered_tune_voltage		double			
rf_tune_readback_voltage		double			
status		RADAR_STATUS_STATES			
mode		RADAR_STATUS_MODES			
current_Furuno_radar		FURUNO_TYPES			
furuno_filter_status		unsigned short			
furuno_stagger_status		unsigned short			
blanking_status		unsigned short			
gp_sensors		unsigned short			
an_ssq_82	0 = Permit Enable, 1 = Deny Permit	unsigned short			
furuno_azimuth_readback		unsigned short			
arp_adjustment		unsigned short			
led_status		unsigned short			
furuno_generic		unsigned short			
currentPort		PORT_SELECTOR			
tx_trig_active		unsigned short			
bearing_pulse_active		unsigned short			
heading_flash_active		unsigned short			
radar_standby		unsigned short			
performance_monitor		unsigned short			
rtPowered		unsigned short			
antenna_rotate_power		unsigned short			
acp_active		unsigned short			
positive_12v		unsigned short			
negative_12v		unsigned short			
arp_active		unsigned short			
host_trigger_active		unsigned short			
interlock_opened		unsigned short			
radar_safe	antenna motor disabled, or safety switch	unsigned short			
generic_radar_bearing_mode	0 = relative, 1 = true	unsigned short			
ft_trigger_active		unsigned short			
mm_true_bearing		unsigned short			
mm_radiate_status		unsigned short			
tune_indicator		unsigned short			
mag_current		unsigned short			
mic_current		unsigned short			
tune_control_feedback_coarse		unsigned short			
tune_control_feedback_fine		unsigned short			
main_bang_supress_level		unsigned short			

main_bang_supress_timing		unsigned short			
magnetron_hours		unsigned short			

Table 56. AMP_RDP_RIPC_SPO(RP_RIPC_SPO_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
error	RIPC: Error Code	long			
targets_scan	RIPC: Number of targets generated per scan	long			
tracks_scan	RIPC: Number of tracks maintained per scan	long			
scan_corr_n	RIPC: Current scan-scan correlator N of N/M	long			
scan_corr_m	RIPC: Current scan-scan correlator M of N/M	long			

Table 57. AMP_RDP_RSC_SPO (RP_RSC_SPO_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
error	RIPC: Error Code	long			
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 58. AMP_RDP_SCAN_STATUS (RP_SCAN_REPORT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
rvp_status	Current RVP reported status	unsigned long			
arp_time	ARP time in seconds from midnight when the scan report message was sent	unsigned long			
targets_scan	Number of targets generated per scan	unsigned long			
tracks_scan	Number of tracks maintained per scan	unsigned long			
scan_corr_n	Current scan-scan correlator N of N/M	unsigned long			
scan_corr_m	Current scan-scan correlator M of N/M	unsigned long			
detections_scan	Number of detections generated per scan	unsigned long			
centroids_scan	Number of centroids generated per scan	unsigned long			
sweeps_behind	Number of sweeps behind in processing	unsigned long			

Table 59. AMP_RDP_SPDP_SPO(RP_SPDP_SPO_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
error	SPDP: Error code	long			
detections_scan	SPDP: Number of detections generated per scan	long			
centroids_scan	SPDP: Number of centroids generated per scan to R IPC	long			
sweeps_behind	SPDP: Number of sweeps behind in processing	long			

Table 60. AMP_RDP_STATUS_OUTPUT (RP_STATUS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
code	use RP_STATUS_CODE	unsigned long			
value	RP status value	unsigned long			
message [RP_STATUS_LENGTH]	RP defined status message; NULL = Clear	char			

Table 61. AMP_ROLE_ASSIGNMENT_TABLE (ICC_ROLE_ASSIGNMENT_TABLE_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
iccMaster	IP Address of ICC Master	int	IP address	N/A	N/A
radarControl [MAX_RP_UNITS]	Radars connected to RP and IP addresses of SPDP running RCC task and of OP with primary control, see header file for the definition of record	RADAR_CONTROL	N/A	N/A	N/A
zweiteImage [MAX_RP_UNITS]	IP addresses of RSC and of OP with secondary control, see header file for the definition of record	ICC_ASSOCIATION	N/A	N/A	N/A
fusion	IP address of RSC running fusion task, see header file for the definition of record	ICC_ASSOCIATION	N/A	N/A	N/A
nsiu	IP addresses of SPDP running NSIU task and of OP with NAV control responsibilities, see header file for the definition of record	ICC_ASSOCIATION	N/A	N/A	N/A

Table 62. AMP_RP_IDENTIFICATION (ICC_RP_ASSOCIATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
radarSiteID	radar site identification	unsigned int	cardinal	0..255	N/A
currentAntenna	radar model identification	int	enum	XN3A, XN4A, XN5A, XN5AF, _mk92, _sps55, _sps67, _fps109, _sps64,	N/A

Table 63. AMP_RSC_IDENTIFICATION (ICC_RSC_ASSOCIATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
radarRPM	scan rate	int			
imageSubtype	CAT code to transmit image as	MESSAGE_SUBTYPE	enum	CAT_RADAR_1_ANY_IMAGE, CAT_RADAR_2_ANY_IMAGE, CAT_RADAR_3_ANY_IMAGE, CAT_RADAR_4_ANY_IMAGE.	N/A
imageSelection	primary or secondary	ICC_RSC_IMAGE	enum	PRIMARY, SECONDARY	N/A

Table 64. AMP_SELECT_AGC_TOGGLE (RP_AGC_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 -7	1
agc	AGC on/off selection	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A

Table 65. AMP_SELECT_AUTO_MODE (RP_AUTO_MODE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
---------------	-------------	-----------	-------	-------	-----------

header	identifies message format	HEADER	N/A	N/A	N/A
range_scale	different range scales	RANGE_SCALES	enum	RANGE_SCALE_POINT_375, RANGE_SCALE_POINT_750, RANGE_SCALE_1_POINT_5, RANGE_SCALE_3_POINT_0, RANGE_SCALE_6_POINT_0, RANGE_SCALE_12_POINT_0, RANGE_SCALE_24_POINT_0, RANGE_SCALE_48_POINT_0, RANGE_SCALE_96_POINT_0	N/A

Table 66. AMP_SELECT_BLANKING (BLANKING_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
blanking_toggle	indicates if blanking is enable/	unsigned short	enum	0 = Disable, 1 = Enable	N/A

Table 67. AMP_SELECT_ENUM_RANGE_SCALE (ENUM_RANGE_SCALE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
range_scale	commanded range scale	RANGE_SCALES	enum	RANGE_SCALE_POINT_375, RANGE_SCALE_POINT_750, RANGE_SCALE_1_POINT_5, RANGE_SCALE_3_POINT_0, RANGE_SCALE_6_POINT_0, RANGE_SCALE_12_POINT_0, RANGE_SCALE_24_POINT_0, RANGE_SCALE_48_POINT_0, RANGE_SCALE_96_POINT_0	N/A

Table 68. AMP_SELECT_FILTERING (FILTER_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
filter_toggle	indicates if the RCC low pass filters is enable	unsigned short	enum	0 = Disable, 1 = Enable	N/A

Table 69. AMP_SELECT_IR_FALL (IR_FALL_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
ir_fall		unsigned short	TBD	TBD	TBD
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 -7	1

Table 70. AMP_SELECT_IR_RISE (IR_RISE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
ir_rise		unsigned short	TBD	TBD	TBD
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 -7	1

Table 71. AMP_SELECT_LANDMASK_VIEW (RP_LAND_MASK_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
view_land_mask	Land mask: use ENABLE/DISABLE	unsigned long	enum	0 = Disable, 1 = Enable	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 -7	1

Table 72. AMP_SELECT_MANUAL_MODE (RP_MANUAL_MODE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
mk_92_pw		MK92_PWS	enum	MK92_PW22, MK92_PW40, MK92_PW53	N/A
prf		unsigned short	Hz	600 - 2200	1 Hz

Table 73. AMP_SELECT_NEAR_LAND (NEAR_LAND_MODE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
near_land_mode_select		unsigned short	TBD	TBD	TBD
pad		unsigned short	N/A	N/A	N/A

Table 74. AMP_SELECT_ORIENTATION (RP_SELECT_ORIENTATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
display_orientation	NORTH UP, HEAD UP, COURSE UP, TRUE MOTION	int	enum	0 = North Up, 1 = Head Up, 2 = Course Up, 3 = True Motion	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 75. AMP_SELECT_PM (PERFORMANCE_MONITOR_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
pm_select	indicates if Furuno performance monitor is enable	unsigned short	enum	0 = Disable, 1 = Enable	N/A

Table 76. AMP_SELECT_PULSE_IR (RP_PULSE_IR_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
interference_rejection	interference rejection to set for selected radar, use STATUS_ON_OR_OFF	unsigned long	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 77. AMP_SELECT_RADAR_STATE (RP_STATE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
radar_state	identifies radar command states	RADAR_CMD_STATES	enum	RADAR_OFF, STANDBY, RADIATE, ROTATE, ROTATE_RADIATE	N/A

Table 78. AMP_SELECT_RANGE_SCALE (RP_RANGE_SCALE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
range_scale	Radar range scale (meters) if RP_SCAN_CONVERTER, Tracker range (nm) if RP_TARGET_EXTRACTOR	float	TBD	TBD	TBD
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 79. AMP_SELECT_STC_TOGGLE (RP_STC_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
stc_curve [stc_state]		RP_STC_CURVE	enum	ENABLED, DISABLED	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 80. AMP_SELECT_SYSTEM_TRACK_AFFIL (SYSTEM_TRACK_AFFIL_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify msg. type	HEADER	N/A	N/A	N/A
class	Classification	unsigned short	enum	AFFIL_UNKNOWN, AFFIL_FRIENDLY, AFFIL_HOSTILE, AFFIL_NEUTRAL, AFFIL_OWNSHIP	N/A
cti	Sys. track number (1-1000)	unsigned short	cardinal	1 - 1000	N/A

tsi	Fusion source identification	unsigned int	cardinal	1 - 255	N/A
-----	------------------------------	--------------	----------	---------	-----

Table 81. AMP_SELECT_SYSTEM_TRACK_CLASS (SYSTEM_TRACK_CLASS_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify mesg. type	HEADER	N/A	N/A	N/A
tsi	Fusion source identification	unsigned int	cardinal	1 - 255	N/A
cti	System track number	unsigned short	cardinal	1 - 1000	N/A
class	Assigned Classification (see Table 82)	unsigned short	enum	CLASS_TYPE	N/A

Table 82. AMP_SELECT_TRACK_CLASS (SYSTEM_TRACK_CLASS_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify mesg. type	HEADER	N/A	N/A	N/A
class	Classification	unsigned short	enum	CLASS_UNKNOWN, CLASS_MERCHANT, CLASS_CARGO, CLASS_SRCH_CTR, CLASS_NAVY, CLASS_TUG, CLASS_FERRY, CLASS_PASSENGER, CLASS_COAST_GUARD, CLASS_HAZMAT, CLASS_TOWING, CLASS_FISHING, CLASS_DRIFTER, CLASS_DREDGE, CLASS_TRAWLER, CLASS_LEISURE, CLASS_HOVERCRAFT, CLASS_MARINE_LIFE, CLASS_ICE_BERG, CLASS_DITCHED_AIRCRAFT, CLASS_MAN_IN_WATER, CLASS_DISTRESSED_VESSEL, CLASS_OIL_PLATFORM, CLASS_AIR_HELO, CLASS_AIR_FIXEDWING	N/A
cti	Sys. track number (1-1000)	unsigned short	cardinal	1 - 1000	N/A
tsi	Fusion source identification	unsigned int	cardinal	1 - 255	N/A

Table 83. AMP_SELECT_VIDEO_SOURCE (RCC_VIDEO_SELECTION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A

videoSource	identifies video source, see header file for definition of record	RCC_VIDEO_SOURCE	N/A	N/A	N/A
-------------	---	------------------	-----	-----	-----

Table 84. AMP_SENSOR_TRACK_DROP (TRACK_DROP_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
messagetype	Reserved space to identify message type	HEADER	N/A	N/A	N/A
cti	Contact Track Identification	unsigned short	cardinal	1 - 1000	N/A
tsi	Track Source Identification	unsigned int	cardinal	1 - 255	N/A
time	Contact message timestamp (in seconds since midnight)	double	seconds	finite	1 microsecond
vts_cti	VTS label	long			

Table 85. AMP_SENSOR_TRACK_LOST (TRACK_LOST_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
messagetype	Reserved space to identify message type	HEADER	N/A	N/A	N/A
cti	Contact Track Identification	unsigned short	cardinal	1 - 1000	N/A
tsi	Track Source Identification	unsigned int	cardinal	1 - 255	N/A
time	Contact message timestamp (in seconds since midnight)	double	seconds	finite	1 microsecond
vts_cti	VTS label	long			

Table 86. AMP_SENSOR_TRACK_REPORT (SENSOR_TRACK_TYPE)

Message Field	Description	Data Type	Units	Range	Precision
ownship_x_pos	ownship x for true motion processing (meters)	double	meters	N/A	N/A
ownship_y_pos	ownship y for true motion processing (meters)	double	meters	N/A	N/A
display_modes_options	(1) : head_up (2) : head_up, north_up (3) : head-up, north_up, course_up	int	enum	1 - 3	N/A
time	Contact message timestamp (in seconds since 1970)	double	seconds	finite	1 microsecond
cti	Contact Track Identification	unsigned short	cardinal	1 - 1000	N/A
tsi	Track Source Identification	unsigned short	cardinal	1 - 255	N/A
x_pos	Track X position (meters)	double	meters	0 - 177, 792	1 meter
y_pos	Track Y position (meters)	double	meters	0 - 177, 792	1 meter
azimuth_extent	Radar measured azimuth extent of track (meters)	unsigned short	degrees	180	1 degree
range_extent	Radar measured range extent of track (meters)	unsighed short	meters	0 - 999	1 meter
course_est	RP course estimate (degrees)	float	degrees	0.0 - 359.9	0.1 degree
speed_est	RP speed estimate (knots)	float	knots	0.0 - 210.0	0.1 knot
quality	Contact Quality	unsigned short	cardinal	1 - 9	N/A
class	Contact classification	CLASS_TYPE	enum	CLASS_UNKNOWN, CLASS_MERCHANT, CLASS_CARGO, CLASS_SRCH_CTR, CLASS_NAVY, CLASS_TUG, CLASS_FERRY, CLASS_PASSENGER, CLASS_COAST_GUARD, CLASS_HAZMAT, CLASS_TOWING, CLASS_FISHING, CLASS_DRIFTER, CLASS_DREDGE, CLASS_TRAWLER, CLASS_LEISURE, CLASS_HOVERCRAFT, CLASS_MARINE_LIFE, CLASS_ICE_BERG, CLASS_DITCHED_AIRCRAFT, CLASS_MAN_IN_WATER, CLASS_DISTRESSED_VESSEL, CLASS_OIL_PLATFORM, CLASS_AIR_HELLO, CLASS_AIR_FIXEDWING	N/A
length	RP guess at length of track (meters)	short	meters		1 meter
beam	RP guess at beam of track (meters)	short	meters		1 meter
label [LABEL_SIZE]	Contact name	char			
bearing_est	RP smoothed/filtered bearing estimate (degrees)	float	degrees	0.0 - 359.9	0.1 degree
range_est	RP range estimate (meters)	unsigned long	meters		1 meter
radar_range	Radar measured range (meters)	unsigned long	meters		1 meter
radar_bearing	Radar measured bearing (degrees)	float	degrees	0.0 - 359.9	0.1 degree
acquisition_id	Track acquisition indicator: see TRACK_ACQUISITION enum	unsigned long	enum	AUTO_TRACK_ACQUISITION, MANUAL_TRACK_ACQUISITION,	N/A

				LOCAL_TRACK_ACQUISITION, LOST_TRACK_ACQUISITION, LAST_TRACK_ACQUISITION	
affil	Contact affiliation	AFFILIATION_TYPE	enum	AFFIL_UNKNOWN, AFFIL_FRIENDLY, AFFIL_HOSTILE, AFFIL_NEUTRAL, AFFIL_OWNSHIP	N/A
coast	Coast track indicator, set indicates this is a coast track	unsigned short	cardinal	0, 1	N/A
az_extent	Azimuth extent (degrees)	float	degrees	180	1 degree
vessels	Number of vessels/barges (in tow)	unsigned long			
vts_cti	VTS label	long			
num_id_links	Number of id links in the system track num_id_links = 1 : unfused num_id_links > 1 : fused	short	cardinal	1 - 5	N/A

Table 87. AMP_SET_ACP_THRESHOLD (RP_ACP_THRESHOLD_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1
threshold	threshold (3-30 V)	float	volts	3 - 30	1 volt

Table 88. AMP_SET_ARP_THRESHOLD (RP_ARP_THRESHOLD_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1
threshold	threshold (3-30 V)	float	volts	3 - 30	1 volt

Table 89. AMP_SET_AUTO_RF_TUNE (RF_TUNING_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
rf_tuning_toggle	identifies if auto tune is disable	unsigned short	enum	0 = Disable, 1 = Enable	N/A

Table 90. AMP_SET_BIAS_CORRECTION (RP_AZIMUTH_BIAS_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
azimuth	azimuth correction factor applied to radar data (deg)	float	degrees	0.0 - 359.9	0.1 degree
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 91. AMP_SET_DECAY_RATE (RP_DECAY_RATE_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
freeze	Use values: DISABLE, ENABLE	unsigned long	enum	0 = Disable, 1 = Enable	N/A
persistence	Number of antennae scans or history trails	unsigned long	scans	1 - 127	1
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 92. AMP_SET_FTC (RP_FTC_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
ftc_state	FTC: use ENABLE/DISABLE	unsigned long	enum	0 = Disable, 1 = Enable	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1
ftc	actual RDE/RSC FTC	unsigned long	ns	50 - 1800	10

Table 93. AMP_SET_GUARD_CELLS (RP_GUARD_CELLS_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
guard_cells	# of guard cells (2-64, even numbers only)	unsigned long	cardinal	2 - 64	2

Table 94. AMP_SET_MANUAL_RF_TUNE (RF_TUNING_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
rf_tuning_coarse	RCC DAC input value	unsigned short	int	0 - 255	1
rf_tuning_fine	RCC DAC input value	unsigned short	int	0 - 255	1

Table 95. AMP_SET_MISSION (RP_MISSION_MODE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
mission_selected		MISSION_MODES	enum	BUOY, SHIP, MM_SHORT, MM_LONG, CRUSING, HARBOR, COAST, OCEAN, ROUGH	N/A
mode		MISSION_RANGE_SCALE_MODE	enum	PW_PRF_MISSION, PW_PRF_AUTO	N/A
pw		unsigned short	nanoseconds	80 - 1200	1 nanosecond
prf		unsigned short	Hertz	600 - 2200	1 Hertz
os_cfar_rank		unsigned short			
os_cfar_guard		unsigned short			
min_cluster_size_in_sweeps		unsigned short			
range_gap		unsigned short			
ftc		FTC_MODE	enum	FTC_OFF, FTC_AUTO, FTC_MANUAL	N/A
max_speed		unsigned short			
growth		unsigned short			

Table 96. AMP_SET_ORIGIN_OFFSET (RP_ORIGIN_OFFSET_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
x_offset	offset in x-direction in meters from origin, precision depends on range scale selected	float	meters	0 - 177792	1.36 to 347.25
y_offset	offset in y-direction in meters from origin, precision depends on range scale selected	float	meters	0 - 177792	1.36 to 347.25
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 97. AMP_SET_RADAR_IMAGE (RADAR_IMAGE_SELECT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
radar_image_select	primary or secondary image	unsigned short	N/A	N/A	N/A

Table 98. AMP_SET_RADAR_MAIN_BANG (MAIN_BANG_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
main_bang_level	RCC DAC input value	unsigned short	int	0 - 255	1
main_bang_timing	RCC DAC input value	unsigned short	int	0 - 255	1

Table 99. AMP_SET_RADAR_PARAMETERS (RP_PARAMETERS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
radarInfo	AMP_SET_RADAR_PARAMETERS, see header file for definition of record	RADAR_ADT	N/A	N/A	N/A
radarMode	AMP_SELECT_AUTO_MODE	RADAR_MODES	enum	MANUAL, MISSION, AUTO	N/A
signalThreshold	AMP_SET_RADAR_PARAMETERS, see header file for definition of record	SIGNAL_THRESHOLDS	N/A	N/A	N/A
clutterSuppression	AMP_SET_RADAR_PARAMETERS, see header file for definition of record	CLUTTER_SUPPRESSION	N/A	N/A	N/A
displayRangeScale	AMP_SELECT_RANGE_SCALE	RANGE_SCALES	enum	RANGE_SCALE_POINT_375, RANGE_SCALE_POINT_750, RANGE_SCALE_1_POINT_5, RANGE_SCALE_3_POINT_0, RANGE_SCALE_6_POINT_0, RANGE_SCALE_12_POINT_0, RANGE_SCALE_24_POINT_0, RANGE_SCALE_48_POINT_0, RANGE_SCALE_96_POINT_0	N/A
videoBias	AMP_SET_VIDEO	double	TBD	TBD	TBD
videoGain	AMP_SET_VIDEO	double	TBD	TBD	TBD
agcState	AMP_SELECT_AGC_TOGGLE	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A
guardCells	AMP_SET_GUARD_CELLS	unsigned long	TBD	TBD	TBD
rvfScalar	AMP_SET_SCALER_VALUE	double	TBD	TBD	TBD
rangeBiasAdjust	AMP_SET_RANGE_BIAS	double	TBD	TBD	TBD
trackingRange	AMP_SELECT_RANGE_SCALE	double	TBD	TBD	TBD
correlationScanCount	AMP_SET_TRACK_SCAN_COUNT	unsigned long	TBD	TBD	TBD
minSpeedFilter	AMP_SET_SENSOR_TRACK_SPEEDS	unsigned long	TBD	TBD	TBD
maxSpeedFilter	AMP_SET_SENSOR_TRACK_SPEEDS	unsigned long	TBD	TBD	TBD

Table 100. AMP_SET_RANGE_BIAS (RP_RANGE_BIAS_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
range_bias		unsigned long	TBD	TBD	TBD

Table 101. AMP_SET_RCC_PARAMETERS (RCC_PARAMETERS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
---------------	-------------	-----------	-------	-------	-----------

header	identifies message format	HEADER	N/A	N/A	N/A
videoSource	AMP_SELECT_VIDEO_SOURCE	RCC_VIDEO_SOURCE	N/A	N/A	N/A
furunoPort	connected Furuno via Jbox	PORT_SELECTOR	enum	1 = primary port, 2 = secondary port	N/A
radarInfo	AMP_SET_RADAR_PARAMETERS	RADAR_ADT	N/A	N/A	N/A
radarMode	AMP_SET_MISSION	RADAR_MODES	enum	manual, mission, auto	N/A
radarState	AMP_SELECT_RADAR_STATE	RADAR_CMD_STATES	enum	RADAR_OFF, STANDBY, RADIATE, ROTATE, ROTATE_RADIATE	N/A
stagger	AMP_SELECT_STAGGER	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A
rangeScaleDetent	AMP_SELECT_ENUM_RANGE_SCALE	RANGE_SCALES	enum	range_scale_point_375, range_scale_point_750, range_scale_1_point_5, range_scale_3_point_0, range_scale_6_point_0, range_scale_12_point_0, range_scale_24_point_0, range_scale_48_point_0, range_scale_96_point_0	N/A
missionSelected		MISSION_MODES	enum	BUOY, SHIP, MM_SHORT, MM_LONG, CRUSING, HARBOR, COAST, OCEAN, ROUGH	N/A
missionRangeScaleMode		MISSION_RANGE_SCALE_MODE	enum	PW_PRF_MISSION, PW_PRF_AUTO	N/A
mission PW		int	ns	80 - 1200	10
missionPRF		int	Hz	600 - 2200	100
MBsuppressionLevel	AMP_SET_MAIN_BANG	unsigned short	N/A	0 - 255	1
MBtimingLevel	AMP_SET_MAIN_BANG	unsigned short	N/A	0 - 255	1
rfTuneMode	AMP_SET_AUTO_RF_TUNE	unsigned short	enum	0 = Disable, 1 = Enable	N/A
coarseTuneValue	AMP_SET_MANUAL_RF_TUNE	unsigned short	int	0 - 255	1
fineTuneValue	AMP_SET_MANUAL_RF_TUNE	unsigned short	int	0 - 255	1
lowPassFilter	AMP_SELECT_FILTERING	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A

performanceMonitor	AMP_SELECT_PM	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A
blankingMode	AMP_SELECT_BLANKING	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A

Table 102. AMP_SET_RP_PARAMETERS (RP_PARAMETERS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
radarInfo	AMP_SET_RADAR_PARAMETERS	RADAR_ADT	N/A	N/A	N/A
radarMode	AMP_SELECT_AUTO_MODE	RADAR_MODES	enum	MANUAL, MISSION, AUTO	N/A
signalThreshold	AMP_SET_RADAR_PARAMETERS	SIGNAL_THRESHOLDS	N/A	N/A	N/A
clutterSuppression	AMP_SET_RADAR_PARAMETERS	CLUTTER_SUPPRESSION	N/A	N/A	N/A
displayRangeScale	AMP_SELECT_RANGE_SCALE	RANGE_SCALES	enum	RANGE_SCALE_POINT_375, RANGE_SCALE_POINT_750, RANGE_SCALE_1_POINT_5, RANGE_SCALE_3_POINT_0, RANGE_SCALE_6_POINT_0, RANGE_SCALE_12_POINT_0, RANGE_SCALE_24_POINT_0, RANGE_SCALE_48_POINT_0, RANGE_SCALE_96_POINT_0	N/A
videoBias	AMP_SET_VIDEO	double	TBD	TBD	TBD
videoGain	AMP_SET_VIDEO	double	TBD	TBD	TBD
agcState	AMP_SELECT_AGC_TOGGLE	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A
guardCells	AMP_SET_GUARD_CELLS	unsigned long	TBD	TBD	TBD
rvfScalar	AMP_SET_SCALER_VALUE	double	TBD	TBD	TBD
rangeBiasAdjust	AMP_SET_RANGE_BIAS	double	TBD	TBD	TBD
trackingRange	AMP_SELECT_RANGE_SCALE	double	TBD	TBD	TBD
correlationScanCount	AMP_SET_TRACK_SCAN_COUNT	unsigned long	TBD	TBD	TBD
minSpeedFilter	AMP_SET_SENSOR_TRACK_SPEEDS	unsigned long	TBD	TBD	TBD
maxSpeedFilter	AMP_SET_SENSOR_TRACK_SPEEDS	unsigned long	TBD	TBD	TBD

Table 103. AMP_SET_RSC_PARAMETERS (RSC_PARAMETERS_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
primaryRSC	facilitates selection of RSC	BOOL	enum	FALSE,	N/A

				TRUE	
radarInfo	AMP_SET_RSC_PARAMETERS	RADAR_ADT	N/A	N/A	N/A
thresholdState	AMP_SET_THRESHOLD_FORMAT	unsigned	enum	0 = RP_AUTO_MODE, 1 = RP_MANUAL_MODE	N/A
thresholdLevel	AMP_SET_THRESHOLD_FORMAT	unsigned	N/A	0 - 63	1
signalThreshold	AMP_SET_RSC_PARAMETERS	SIGNAL_THRESHOLDS	N/A	N/A	N/A
clutterSuppression	AMP_SET_RSC_PARAMETERS	CLUTTER_SUPPRESSION	N/A	N/A	N/A
imageFormat	AMP_RADAR_IMAGE_PRESENTATION	RP_PRESENTATION_TYPE	enum	X_Y_FORMAT, RHO_THETA_FORMAT	N/A
imageDensity	AMP_SET_THRESHOLD_FORMAT	RP_RADAR_IMAGE_BITS	enum	IMAGE_2_BIT, IMAGE_4_BIT	N/A
bearingIndictor	AMP_GENERIC_RADAR_ORIENTATION	RP_AZIMUTH_ORIENTATION	enum	HEAD_RELATIVE, NORTH_RELATIVE	N/A
yOffset	AMP_SET_ORIGIN_OFFSET	double	meters	0 - 177792	1.36 to 347.25
xOffset	AMP_SET_ORIGIN_OFFSET	double	meters	0 - 177792	1.36 to 347.25
rangeScale	meters;AMP_SELECT_RANGE_SCALE	double	meters	0 - 177792	1.36 to 347.25
videoBias	AMP_SET_VIDEO	double	volts	-0.7 - +0.7	0.01
videoGain	AMP_SET_VIDEO	double	%	0 - 100	0.01
agcState	AMP_SELECT_AGC_TOGGLE	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A
freezeImage	AMP_SET_DECAY_RATE	BOOL	enum	FALSE, TRUE	N/A
persistence	AMP_SET_DECAY_RATE	unsigned	scans	0 - 127	1
graphic_ID	AMP_LAND_MASK_SELECT	GEO_ID	enum	GEO_ID_LAND, GEO_ID_ANCHORAGE, GEO_ID_WATER_AREA, GEO_ID_AUTO_ACQUIRE_1, GEO_ID_AUTO_ACQUIRE_2, GEO_ID_AUTO_ACQUIRE_3, GEO_ID_AUTO_ACQUIRE_4, GEO_ID_AUTO_ACQUIRE_5, GEO_ID_AUTO_ACQUIRE_6, GEO_ID_AUTO_ACQUIRE_7, GEO_ID_AUTO_ACQUIRE_8	N/A
mapState	AMP_LAND_MASK_SELECT	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A
viewLandMask	AMP_SELECT_LANDMASK_VIEW	STATUS_ON_OR_OFF	enum	STATUS_OFF, STATUS_ON, STATUS_UNKNOWN	N/A

Table 104. AMP_SET_SCALER_VALUE (RP_RVF_SCALER_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A

rvf_scaler		float	TBD	TBD	TBD
------------	--	-------	-----	-----	-----

Table 105. AMP_SET_SENSOR_TRACK_SPEEDS (TRACK_SPEED_FILTER_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
minspeed	Minimum speed for acquiring a track (Meters/Sec)	unsigned long	TBD	TBD	TBD
maxspeed	Maximum speed for acquiring a track (Meters/Sec)	unsigned long	TBD	TBD	TBD
speed_bin_size	Size of speed bin in Scan/Scan Correlator (m/s)	double	TBD	TBD	TBD

Table 106. AMP_SET_STC_CURVE (RP_STC_SET_CURVE_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
stc_curve [RP_STC_ENTRIES]	see header file for definition of record	RP_STC_CURVE	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1

Table 107. AMP_SET_SYSTEM_TIME (SYSTEM_TIME_SET_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
seconds	Seconds and microseconds represent time	unsigned long	seconds	0 - 2147483647	1 second
microseconds	accumulated since Jan 1, 1970	unsigned long	usecond	0 - 1000000	1 usecond

Table 108. AMP_SET_SYSTEM_TRACK_LABEL (SYSTEM_TRACK_LABEL_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify mesg. type	HEADER	N/A	N/A	N/A
label [LABEL_SIZE]	Alphanumeric characters that identify a track	char	ASCII	1 - 11	N/A
cti	Sys. track number (1-1000)	unsigned short	cardinal	1 - 1000	N/A
tsi	Fusion source identification	unsigned int	cardinal	1 - 255	N/A

Table 109. AMP_SET_TBM_BIASES (RP_TBM_BIASES_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned short	cardinal	1 - 7	1
Boases [RP_MAX_TBM_CHOICES]	Up to 16 possible TBM choices, positive or negative values	long	TBD	TBD	TBD

Table 110. AMP_SET_THRESHOLD_FORMAT (RP_THRESHOLD_FORMAT_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	identifies message format	HEADER	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1
op_intensity	Use values: RP_INTENSITY_2_BIT, RP_INTENSITY_4_BIT	unsigned long	enum	0 = RP_INTENSITY_1_BIT 1 = RP_INTENSITY_2_BIT 2 = RP_INTENSITY_3_BIT 3 = RP_INTENSITY_4_BIT	N/A
op_threshold	Use values: RP_AUTO_MODE, RP_MANUAL_MODE	unsigned long	enum	0 = RP_AUTO_MODE, 1 = RP_MANUAL_MODE	N/A
video_threshold	Video Threshold (levels 1 to 31)	unsigned long	N/A	0 - 63	1

Table 111. AMP_SET_TRACK_SCAN_COUNT (TRACK_SCAN_COUNT_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
scans	Number of scans before dropping a coasted track	unsigned long	scans	finite	1 scan

Table 112. AMP_SET_TRIG_THRESHOLD(RP_TRIG_THRESHOLD_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1
threshold	trigger threshold (3-30 V)	float	volts	3 - 30	1 volt

Table 113. AMP_SET_VIDEO (RP_VIDEO_SET_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
equipment_code	Bit Field (RP_SCAN_CONVERTER = 1, RP_TARGET_EXTRACTOR = 2, RP_SEC_SCAN_CONVERTER = 4)	unsigned long	cardinal	1 - 7	1
bias	bias (-.7 - +.7 V)	float	volts	-0.7 - 0.7	0.1 volt
gain	gain (1 - 3 V)	float	volts	1 - 3	1 volt

Table 114. AMP_SETUP_DATA_1 (ICC_SETUP_DATA_MSG_1)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
rt_params [MAX_SSR_RADARS]	radar configuration data, see header file for definition of record	RT_PARAMETERS	N/A	N/A	N/A

Table 115. AMP_SETUP_DATA_2 (ICC_SETUP_DATA_MSG_2)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
rde_params [MAX_STATIONS]	rde configuration data, see header file for definition of record	RDE_RSC_ADJUST	N/A	N/A	N/A
nav_params	nsui configuration data, see header file for definition of record	NAV_PARAMETERS	N/A	N/A	N/A

Table 116. AMP_SIU_STATUS_CONTROL_REQUEST (SIU_STATUS_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Msg header	HEADER	N/A	N/A	N/A
siu_stat_ctrl_cmd	see header file for definition of record	SIU_STATUS_CONTROL_RECORD	N/A	N/A	N/A
siu_stat_ctrl_data	Data related to cmd (heading,speed)	float	degrees/knot	0.0 - 359.99	0.01 degree/knot
siu_stat_ctrl_msg_cnt	Number of lines in following msg array	long	lines	0 - 6	N/A
siu_stat_ctrl_msgs [6] [48]		char	strings	N/A	N/A

Table 117. AMP_SIU_STATUS_CONTROL_RESPONSE (SIU_STATUS_CONTROL_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Msg header	HEADER	N/A	N/A	N/A
siu_stat_ctrl_cmd		SIU_STATUS_CONTROL_RECORD	N/A	N/A	N/A
siu_stat_ctrl_data	Data related to cmd (heading,speed)	float	degrees/knot	0.0 - 359.99	0.01 degree/knot
siu_stat_ctrl_msg_cnt	Number of lines in following msg array	long	lines	0 - 6	N/A
siu_stat_ctrl_msgs [6] [48]		char	strings	N/A	N/A

Table 118. AMP_SPDP_IDENTIFICATION (ICC_SPDP_ASSOCIATION_MSG)

Message Field	Description	Data Type	Units	Range	Precision
message	identifies message format	HEADER	N/A	N/A	N/A
rpcNode	IP address of RPC processor	unsigned int	N/A	N/A	N/A

Table 119. AMP_SYSTEM_ALERT (SYSTEM_ALERT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
alert_id	Alert msg id (see file / ssr/op/common/mmi_utils.h & co.h	int			

Table 120. AMP_SYSTEM_OWNSHIP_REPORT (SYSTEM_OWNSHIP_REPORT_MSG)

Message Field	Description	Data Type	Units	Range	Precision
header	Msg header	HEADER	N/A	N/A	N/A
report_source	Report source identification	int	N/A		N/A
[LABEL_SIZE]	Ship's name	char label	N/A	0 - 9, A - Z	N/A
nav_data	Ship's NAV data information	NAV_DATA_RECORD, see header file for definition of record	N/A	N/A	N/A
nav_current_source	Ship's NAV Cur. source info.	NAV_SOURCE_RECORD, see header file for definition of record	N/A	N/A	N/A
nav_alternate_source	Ship's NAV Alt. source info.	NAV_SOURCE_RECORD, see header file for definition of record	N/A	N/A	N/A
nav_current_status	Ship's NAV Cur. sensor status	NAV_STATUS_RECORD, see header file for definition of record	N/A	N/A	N/A
nav_alternate_status	Ship's NAV Alt. sensor status	NAV_STATUS_RECORD, see header file for definition of record	N/A	N/A	N/A
nav_sigma	Ship's NAV error information	NAV_SIGMA_RECORD, see header file for definition of record	N/A	N/A	N/A
nav_fixed	Fixed offset data	NAV_FIXED_REF_RECORD, see header file for definition of record	N/A	N/A	N/A
nav_env	Ship's NAV environment info.	NAV_ENV_RECORD, see header file for definition of record	N/A	N/A	N/A

Table 121. AMP_SYSTEM_TRACK_DROP (SYSTEM_TRACK_DROP_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
cti	Contact Track Identification	unsigned short	cardinal	1 - 1000	N/A
tsi	Track Source Identification	unsigned int	cardinal	1 - 255	N/A
systemTrackStatus	LOST_SYSTRK DROP_SYSTRK DROP_SYSTRK_NOTF	DROP_SYS_STATUS	enum	0 - 2	N/A

Table 122. AMP_SYSTEM_TRACK_REPORT (SYSTEM_TRACK_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify mesg. type	HEADER	N/A	N/A	N/A
num_of_tracks		int	tracks	1 - 10	1 track
systemTracks [SYSTEM_TRACK_TYPE	N/A	N/A	N/A
MAX_TRACKS_PER_					

RECORD]					
----------	--	--	--	--	--

Table 123. AMP_SYSTEM_TRACK_REPORT cont... (SYSTEM_TRACK_TYPE)

Message Field	Description	Data Type	Units	Range	Precision
ownship_x_pos	fixed reference Point since going into Display Mode option 3	double	meters	N/A	N/A
ownship_y_pos	fixed reference Point since going into Display Mode option 3	double	meters	N/A	N/A
display_modes_options	1 = Head Up; 2 = Head Up, North Up; 3 = Head Up, North Up, Course Up, True Motion	int	cardinal	1 - 3	N/A
time	seconds since midnight 1 January, 1970	double	seconds	finite	1 microsecond
cti	System Track Identifier	unsigned short	cardinal	1-1000	N/A
tsi	Track (Fusion) Source Identification	unsigned int	cardinal	1-255	N/A
x_pos	(smoothed position, meters based on display mode)	double	meters	0 - 177,792	1 meter
y_pos	(smoothed position, meters based on display mode)	double	meters	0 - 177,792	1 meter
azimuth_extent	characteristics of target ship	unsigned short	meters	N/A	1 meter
range_extent	characteristics of target ship	unsigned short	meters	0 - 999	1 meter
course_est	course estimate	float	degrees	0.0 - 359.9	0.1 degree
speed_est	speed estimate	float	knots	0.0 - 210.0	0.1 knot
quality	System track quality 1(low) - 9(high)	unsigned short	cardinal	1 - 9	N/A
class	track classification	CLASS_TYPE (unsigned long)	enum	CLASS_UNKNOWN, CLASS_MERCHANT, CLASS_CARGO, CLASS_SRCH_CTR, CLASS_NAVY, CLASS_TUG, CLASS_FERRY, CLASS_PASSENGER, CLASS_COAST GUARD, CLASS_HAZMAT, CLASS_TOWING, CLASS_FISHING, CLASS_DRIFTER, CLASS_DREDGE, CLASS_TRAWLER, CLASS_LEISURE, CLASS_HOVERCRAFT, CLASS_MARINE_LIFE, CLASS_ICE_BERG, CLASS_DITCHED_AIRCRAFT, CLASS_MAN_IN_WATER, CLASS_DISTRESSED_VESSEL, CLASS_OIL_PLATFORM, CLASS_AIR_HELO, CLASS_AIR_FIXEDWING	N/A
length	Target Length	short	meters	0 - 2000	1 meter
beam	Target Beam	short	meters	0 - 999	1 meter
label[]	alphanumeric characters that identify a track	char	ASCII	1 - 11	N/A
bearing_est	bearing estimate: (if knowing North) North relative	float	degrees	0.0 - 359.9	0.1 degree

	else heading relative				
range_est	range estimate	unsigned long	meters	0 - 177,792	1 meter
radar_range	1852 meters per nautical mile	unsigned long	meters	0 - 177,792	1 meter
radar_bearing	relative to the bridge	float	degrees	0.0 - 359.9	0.1 degree
affil	contact affiliation	AFFILIATION_TYPE	enum	AFFIL_UNKNOWN, AFFIL_FRIENDLY, AFFIL_HOSTILE, AFFIL_NEUTRAL	N/A
num_id_links	number of fused tracks (1 : unfused)	unsigned short	cardinal	1 - 4	N/A
slot_no	(internal usage only)	unsigned short	cardinal	N/A	N/A
acquisition_id	Track acquisition indicator [<i>internal usage</i>] <i>currently not used</i>	TRACK_ACQUISITION	enum	AUTO_TRACK_ACQUISITION, MANUAL_TRACK_ACQUISITION, LOCAL_TRACK_ACQUISITION, LOST_TRACK_ACQUISITION	N/A
board_stat	Vessel boarding status [<i>internal usage</i>]	BOARDING_STATUS_TYPE	enum	VESSEL_NOT_INTEREST, VESSEL_INTEREST, VESSEL_ON_BOARD, VESSEL_BOARDED	N/A
coast	Coast track indicator - <i>currently not used</i>	unsigned short	cardinal		
vessels	Number of vessels/barge (in tow) [<i>internal usage</i>]	unsigned long	cardinal	N/A	N/A

Table 124. AMP_TARGET_DETECTION_REPORT (TARGET_DETECTION_RECORD)

Message Field	Description	Data Type	Units	Range	Precision
messagetype	Reserved space to identify message type	HEADER	N/A	N/A	N/A
ownship_x_pos	w.r.t prime meridian and equator (meters)	double	TBD	TBD	TBD
ownship_y_pos	w.r.t prime meridian and equator (meters)	double	TBD	TBD	TBD
display_modes_options	(1) : head_up (2) : head_up, north_up (3) : head_up, north_up, course_up	int	enum	1 - 3	N/A
deltax	Detection Delta X 1 meter incr	long	TBD	TBD	TBD
deltay	Detection Delta Y 1 meter incr	long	TBD	TBD	TBD
type	Detection type (see DETECTION_TYPE)	DETECTION_TYPE	enum	DETECTION_CONTACT, DETECTION_TARGET	N/A

Table 125. AMP_TARGET_REPORT (TRACK_TARGET_REPORT)

Message Field	Description	Data Type	Units	Range	Precision
header	Reserved space to identify message type	HEADER	N/A	N/A	N/A
x_pos	X position (meters)	long	TBD	TBD	TBD
y_pos	Y position (meters)	long	TBD	TBD	TBD
az_width	Azimuth width (meters)	unsigned short	TBD	TBD	TBD
rng_run_length	Range run length (meters)	unsigned short	TBD	TBD	TBD
aspect_angle	(radians)	float	TBD	TBD	TBD
signal_strength	Peak amplitude	unsigned int	TBD	TBD	TBD
pad	To allow alignment with HPUX compiler	unsigned short	TBD	TBD	TBD

9.

Attachment: RDS API Header Files

The following header files (working snapshot) are attached:

```
5354 Jun 24 14:44 alerts_api.h
4053 Jun 24 14:44 bits_api.h
9707 Jun 24 14:44 graphic_api.h
23847 Jun 24 14:44 iccontrol_api.h
16562 Jun 24 14:44 messages_api.h
8011 Jun 24 14:44 msdbuf_api.h
12953 Jun 24 14:44 nsiu_api.h
10588 Jun 24 14:44 queues_api.h
54317 Jun 24 14:44 radar_api.h
19279 Jun 24 14:44 rds_api.h
31244 Jun 24 14:44 tracks_api.h
```